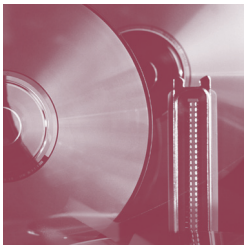
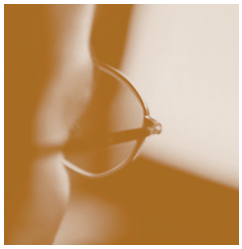
ASG-SmartTest™
Commands Quick Start Guide

Version: 6.0

Publication Number: STX0600-60

Publication Date: February 2002





ASG Worldwide Headquarters Naples, Florida USA | asg.com
1333 Third Avenue South, Naples, Florida 34102 USA Tel: 941.435.2200 Fax: 941.263.3692 Toll Free: 1.800.932.5536

ASG Documentation/Product Enhancement Fax Form

Please FAX comments regarding ASG products and/or documentation to (941) 263-3692.

Company Name	Telephone Number	Site ID	Contact name

Product Name/Publication	Version #	Publication Date
Product:		
Publication:		
Tape VOLSER:		

Enhancement Request:

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)
- Verify whether you received an ASG Service Pack for this product. It may include information to help you resolve questions regarding installation of this ASG product. The Service Pack instructions are in a text file on the distribution media included with the Service Pack.

If You Receive a Voice Mail Message:

1. Follow the instructions to report a production-down or critical problem.
2. Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
3. Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578	941.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.300	49.89.45716.400	support.de@asg.com
Singapore	65.332.2922	65.337.7228	support.sg@asg.com
All other countries:	1.941.435.2200		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578		
Asia	65.332.2922	**Japan/Telecom**	0041.800.9932.5536
Australia	0011.800.9932.5536	**Netherlands**	00.800.3354.3578
Denmark	00.800.9932.5536	**New Zealand**	00.800.9932.5536
France	00.800.3354.3578	**Singapore**	001.800.3354.3578
Germany	00.800.3354.3578	**South Korea**	001.800.9932.5536
Hong Kong	001.800.9932.5536	**Sweden/Telia**	009.800.9932.5536
Ireland	00.800.9932.5536	**Switzerland**	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536	**Thailand**	001.800.9932.5536
Japan/IDC	0061.800.9932.5536	**United Kingdom**	00.800.3354.3578
		All other countries	1.941.435.2200

ASG Web Site

Visit http://www.asg.com, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at http://www.asg.com/asp/emailproductsuggestions.asp.

If you do not have access to the web, FAX your suggestions to product management at (941) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Preface

This *ASG-SmartTest Commands Quick Start Guide* summarizes the command syntax and usage information for the ASG-SmartTest (herein called SmartTest) commands. SmartTest is the Testing/Debugging component of the ASG Existing Systems Workbench (ESW). It automates the time-consuming and error-prone process of testing and debugging application programs. This publication is a guide for installing and maintaining SmartTest.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

The *ASG-SmartTest Commands Quick Start Guide* consists of these chapters:

- Chapter 1, "Command Language Syntax," provides a description and command syntax for the SmartTest commands.
- Chapter 2, "Pseudo Code Statements," describes the pseudo code statements that you can use to insert temporary COBOL code during a test session.
- Chapter 3, "Operand Definitions," defines the command operands.
- Chapter 4, "Program View Line Commands," lists the line commands valid for SmartTest.

- Chapter 5, "Analyze Options," lists the options available to control the output format and to describe COBOL options.
- Chapter 6, "VIASUB/VIASUBDS Parameters," lists the VIASUB/VIASUBDS parameters.
- Chapter 7, "Assigning PF Keys," lists the default and suggested alternate PF key settings.
- Chapter 8, "Storage Area Keywords," describes the storage area keywords for SmartTest, CICS, COBOL II, and system and Assembler.
- Chapter 9, "Action Bar Equivalents To Commands," lists the pull-down and action equivalent to SmartTest commands.

Related Publications

The documentation library for ASG-SmartTest consists of these publications (where *nn* represents the product version number):

- *ASG-Center Installation Guide* (CNX0300-*nn*) contains installation and maintenance information for ASG-Center, the common set of libraries shared by the ASG-ESW suite of products.
- *ASG-SmartTest CICS User's Guide* (STC0200-*nn*) contains specific commands and test session setup information for the CICS environments.
- *ASG-SmartTest for COBOL and Assembler User's Guide* (STA0200-*nn*) contains introductory and usage information for COBOL and Assembler. It also contains test session setup information for the TSO, ISPF, IMS/DB, DB/2, BTS, and Batch environments.

- *ASG-SmartTest IMS User's Guide* (STM0200-*nn*) contains specific commands and test session setup information for the IMS/DC environments.
- *ASG-SmartTest Installation Guide* (STX0300-*nn*) contains information for installing and maintaining ASG-SmartTest.
- *ASG-SmartTest PLI User's Guide* (STL0200-*nn*) contains introductory and usage information about how to use ASG-SmartTest with the PL/I language. It also contains test session setup information for the TSO, ISPF, IMS/DB, DB/2, BTS, and Batch environments.
- *ASG-SmartTest Quick Start for COBOL/ASM* (STA0900-*nn*) summarizes how to use ASG-SmartTest with the COBOL or Assembler language.
- *ASG-SmartTest Quick Start for PL/I* (STL0900-*nn*) summarizes how to use ASG-SmartTest with the PL/I language.
- *ASG-SmartTest Reference Guide* (STX0400-*nn*) contains detailed reference information about CUA pull-downs and pop-ups, ASG-SmartTest command syntax, and pseudo code.
- *ASG-SmartTest Reference Summary* (STX0600-*nn*) summarizes the syntax and usage of ASG-SmartTest commands.
- *ASG-SmartTest TCA User's Guide* (STT0200-*nn*) contains procedures for using the ASG-SmartTest-TCA (Test Coverage Analysis) option.

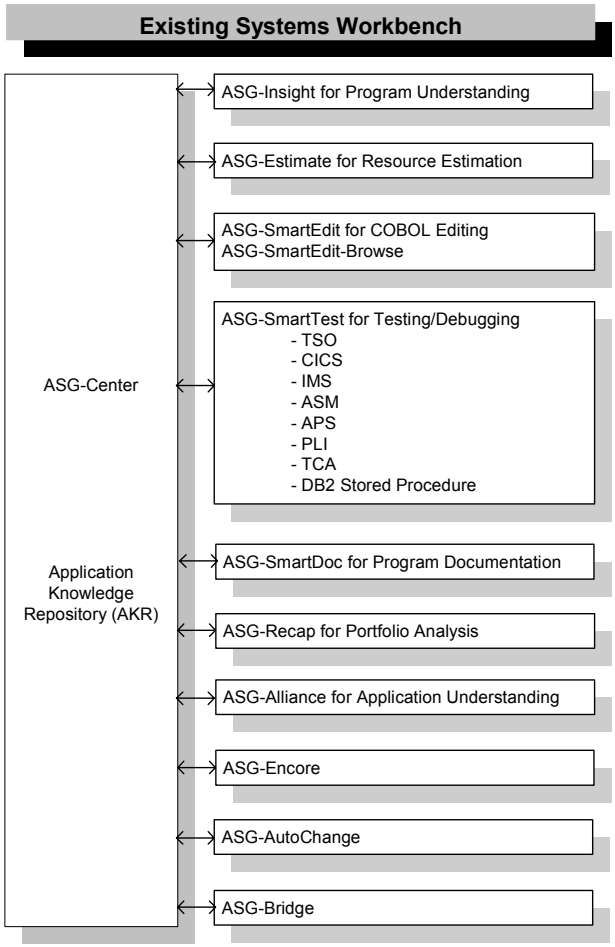
Note:

To obtain a specific version of a publication, contact the ASG Service Desk.

ASG-Existing Systems Workbench (ASG-ESW)

ASG-ESW (herein called ESW) is an integrated suite of components designed to assist organizations in enhancing, redeveloping, or re-engineering their existing systems. ESW products use the Application Knowledge Repository (AKR) to store source program analysis information generated by the Analytical Engine. Figure 1 represents the components of ESW.

Figure 1 • ASG Existing Systems Workbench



This table contains the name and description of each ESW component:

ESW Product	Herein Called	Description
ASG-Alliance	Alliance	The application understanding component that is used by IT professionals to conduct an analysis of every application in their environment. Alliance supports the analysis and assessment of the impact of change requests upon an entire application. Alliance allows the programmer/ analyst to accurately perform application analysis tasks in a fraction of the time it would take to perform these tasks without an automated analysis tool. The impact analysis from Alliance provides application management with additional information for use in determining the resources required for application changes.
ASG-AutoChange	AutoChange	The COBOL code change tool that makes conversion teams more productive by enabling quick and safe changes to be made to large quantities of code. AutoChange is an interactive tool that guides the user through the process of making source code changes.
ASG-Bridge	Bridge	The bridging product that enables field expansion for program source code, without being required to simultaneously expand the fields in files or databases. Because programs are converted in smaller groups, or on a one-by-one basis, and do not require file conversion, testing during the conversion process is simpler and more thorough.

ESW Product	Herein Called	Description
ASG-Center	Center	The common platform for all ESW products. Center provides the common Analytical Engine to analyze the source program and store this information in the AKR. This common platform provides a homogeneous environment for all ESW products to work synergistically.
ASG-Encore	Encore	The program re-engineering component for COBOL programs. Encore includes analysis facilities and allows you to extract code based on the most frequently used re-engineering criteria. The code generation facilities allow you to use the results of the extract to generate a standalone program, a callable module, a complement module, and a CICS server. Prior to code generation, you can view and modify the extracted Logic Segment using the COBOL editor.
ASG-Estimate	Estimate	The resource estimation tool that enables the user to define the scope, determine the impact, and estimate the cost of code conversion for COBOL, Assembler, and PL/I programs. Estimate locates selected data items across an application and determines how they are used (moves, arithmetic operations, and compares). Time and cost factors are applied to these counts, generating cost and personnel resource estimates.

ESW Product	Herein Called	Description
ASG-Insight	Insight	The program understanding component for COBOL programs. Insight allows programmers to expose program structure, identify data flow, find program anomalies, and trace logic paths. It also has automated procedures to assist in debugging program abends, changing a computation, and resolving incorrect program output values.
ASG-Recap	Recap	The portfolio analysis component that evaluates COBOL applications. Recap reports provide function point analysis and metrics information, program quality assessments, intra-application and inter-application comparisons and summaries, and historical reporting of function point and metrics information. The portfolio analysis information can also be viewed interactively or exported to a database, spreadsheet, or graphics package.
ASG-SmartDoc	SmartDoc	The program documentation component for COBOL programs. SmartDoc reports contain control and data flow information, an annotated source listing, structure charts, program summary reports, exception reports for program anomalies, and software metrics.

ESW Product	Herein Called	Description
ASG-SmartEdit	SmartEdit	The COBOL editing component that can be activated automatically when the ISPF/PDF Editor is invoked. SmartEdit provides comprehensive searching, inline copybook display, and syntax checking. SmartEdit allows you to include an additional preprocessor (for example, the APS generator) during syntax checking. SmartEdit supports all versions of IBM COBOL, CICS, SQL, and CA-IDMS.
ASG-SmartTest	SmartTest	The testing/debugging component for COBOL, PL/I, Assembler, and APS programs in the TSO, MVS Batch, CICS (including file services), and IMS environments. SmartTest features include program analysis commands, execution control, intelligent breakpoints, test coverage, pseudo code with COBOL source update, batch connect, disassembled object code support, and full screen memory display.

Invoking ESW Products

The method you use to invoke an ESW product depends on your system setup. If you need assistance to activate a product, see your systems administrator. If your site starts a product directly, use the ISPF selection or CLIST as indicated by your systems administrator. If your site uses the ESW screen to start a product, initiate the ESW screen using the ISPF selection or CLIST as indicated by your systems administrator and then typing in the product command on the command line.

The product names can also vary depending on whether you access a product directly or through ESW. See "ESW Product Integration" on page xvi for more information about using ESW.

To initialize ESW products from the main ESW screen, select the appropriate option on the action bar pull-downs or type the product shortcut on the command line.

Product Name	Shortcut	ESW Pull-down Options
Alliance	AL	Understand ▸ Application
AutoChange	CC	Change ▸ Conversion Set
Bridge	BR	Change ▸ ASG-Bridge
Encore (Re-engineer)	EN	Re-engineer ▸ Program
Estimate	ES	Measure ▸ ASG-Estimate
Insight (Understand)	IN	Understand ▸ Program
Recap (Portfolio Analysis)	RC	Measure ▸ Portfolio
SmartDoc (Document)	DC	Document ▸ Program
SmartEdit	SE	Change ▸ Program **Or** Change ▸ Program with Options
SmartTest	ST	Test ▸ Module/Transaction

ESW Product Integration

Because ESW is an integrated suite of products, you are able to access individual ESW products directly or through the main ESW screen. As a result, you might see different fields, values, action bar options, and pull-down options on a screen or pop-up depending on how you accessed the screen or pop-up.

Certain ESW products also contain functionality that interfaces with other ESW products. Using SmartTest as an example, if Alliance is installed, SmartTest provides a dynamic link to Alliance that can be used to display program analysis information. If Insight is installed and specified during the analyze, the Insight program analysis functions are automatically available for viewing logic/data relationships and execution path. For example, the Scratchpad option is available on the Options pull-down if you have Insight installed. Access to these integrated products requires only that they be installed and executed in the same libraries.

Examples

Example 1. Figure 2 shows the Encore Primary screen that displays when you access Encore directly.

The Encore Primary screen contains these eight action bar menu items: File, View, Extract, Generate, Search, List, Options, and Help.

Figure 2 • Encore Primary Screen

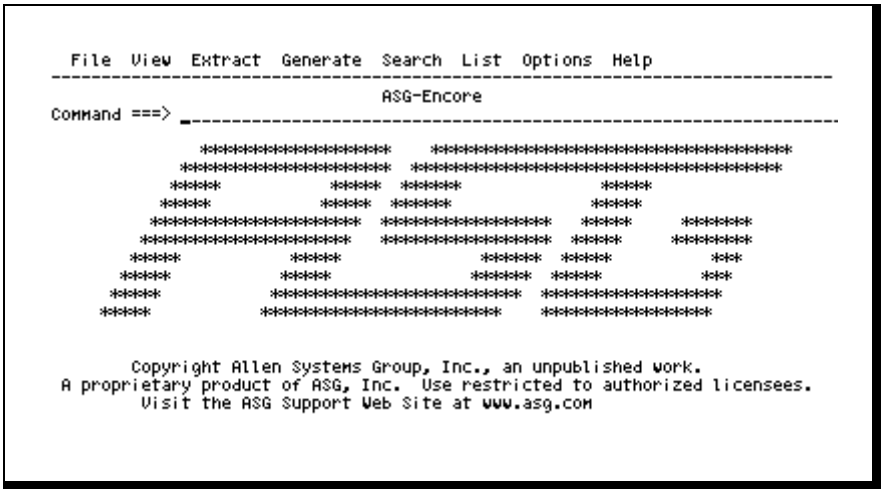
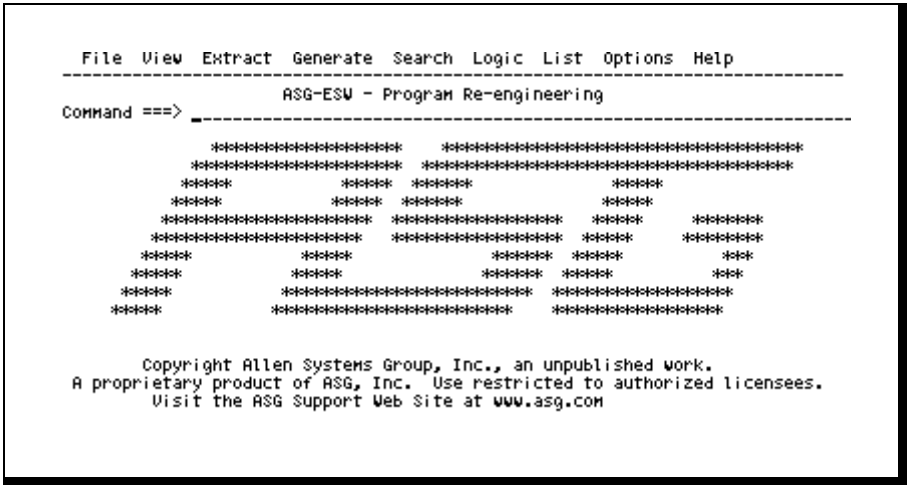


Figure 3 shows the Encore Primary screen that displays when you access Encore through ESW by selecting Re-engineer ▸ Program from the ESW action bar menu. Notice that the Primary screen name changes to ASG-ESW - Program Re-engineering when you enter Encore through ESW. Also, the Logic menu item displays if Insight is installed.

Figure 3 • ESW Encore Primary Screen



Example 2. Figure 4 on page xix shows the File - Analyze Submit pop-up that displays when you access SmartTest directly. Figure 5 on page xix shows the File - Analyze Submit pop-up that displays when you access SmartTest through ESW.

Notice that the Analyze features field in Figure 5 on page xix lists additional ESW products than shown on Figure 4 on page xix. This field is automatically customized to contain the ESW products you have installed on your system.

The actions shown on these screens also vary. For example, the D action (ASG-SmartDoc Options) is available on the File - Analyze Submit screen if the SmartDoc product is installed on your system. In Figure 4, the ASG-SmartDoc Options action is not available.

Figure 4 • File - Analyze Submit Screen

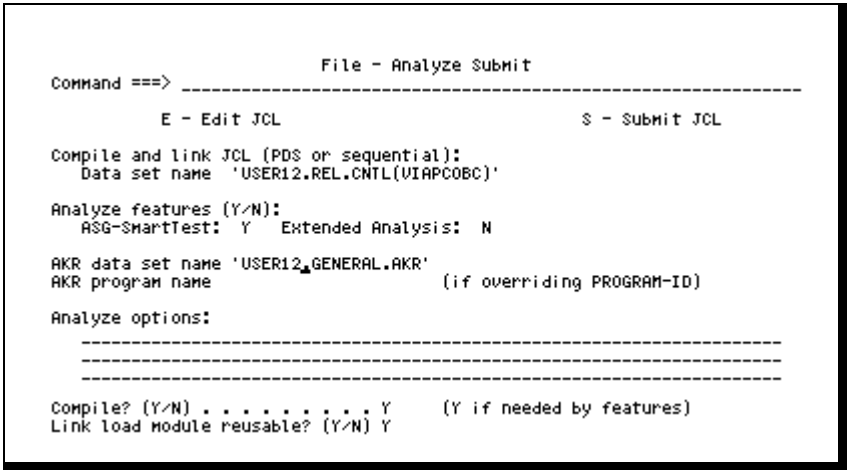
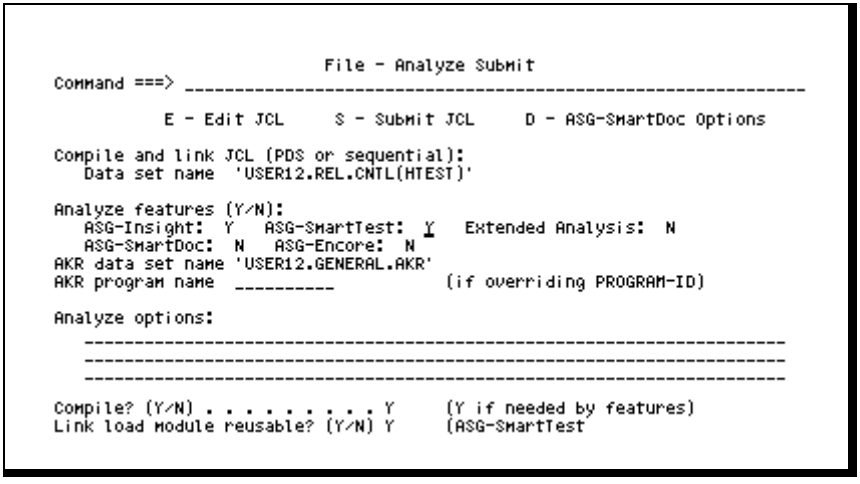


Figure 5 • File - Analyze Submit Screen (Accessed through ESW)

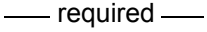


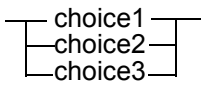
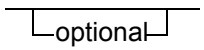
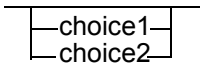
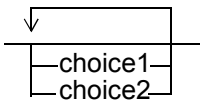
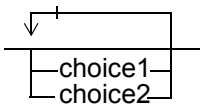
Conventions

The following highlighting conventions are used in this guide:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
lowercase italic monospace	Information that you provide according to your particular situation. For example, you would replace *filename* with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax.
	Also used for denoting brief examples in a paragraph.
Vertical Separator Bar (|) with underline	Options available with the default value underlined (e.g., Y|N).

1 Command Language Syntax

Item	Description
ABBREViations	Illustrates the command abbreviation, which is shown in uppercase letters. Lowercase letters in the command are optional.
lowercase	Indicates user-supplied variable information.
UPPERCASE	Indicates commands or keywords.
Bold	Indicates operands that are available only if SmartTest is installed and a SmartTest analysis has been run on the COBOL program being tested.
Underline	Specifies the default value of an operand.
|	Separates synonymous commands or operands.
	Indicates that the command syntax is continued on the next line.
	Indicates the command syntax is continued from the previous line.
	Indicates the end of the command syntax.
	Indicates that the operand or keyword appearing on the main command line is required.

Item	Description
	Indicates that one operand is required.
	Indicates that an operand or keyword appearing below the main command line is optional.
	Indicates that operands are optional.
	Indicates that more than one operand can be chosen.
	Indicates that operands can be concatenated by placing a plus sign (+) between them.

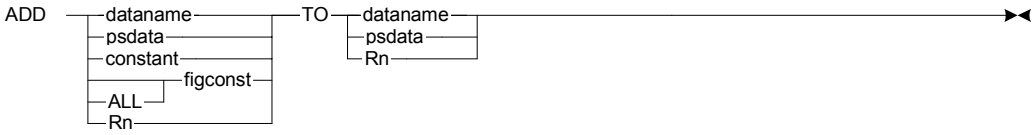
SmartTest supports all ISPF/PDF system commands on the appropriate screens (e.g., UP, DOWN, KEYS). Targets are searched in the order listed. A particular kind of target can be selected by entering the optional target-type prefix (e.g., SUBSET, LABEL, DATA). Data name subordinate operands (e.g., REF, ALIAS) pertain to all datanames in a concatenated series. For COBOL II Release 3 and later programs, a dataname, label name, or program name that may be ambiguous or used multiple times can be qualified with OF.

& (Retain)

Executes the specified primary command and keeps it displayed in the command input area for repeated use or modification.

ADD

Adds the value contained in or represented by the first operand to the specified data item. The value is converted to the proper format for the data item.

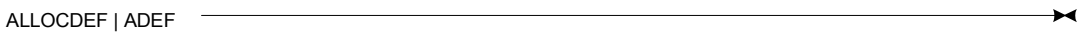


ALLIANCE

Displays the SmartTest/Alliance Interface pop-up used to configure a dynamic link to Alliance. After entering appropriate information in the SmartTest/Alliance Interface pop-up and pressing Enter, SmartTest activates Alliance and runs the script specified on the Query Name field of the SmartTest/Alliance Interface pop-up.

ALLOCDEF

Displays the Options - Product Allocations pop-up used to specify the DASD volumes for the Log, List, Punch, and Work files; and to specify space for the Work file.



ANALYZE

Displays the File - Analyze Submit pop-up that can be used to submit a compile/analyze job without actually ending the current SmartTest function.

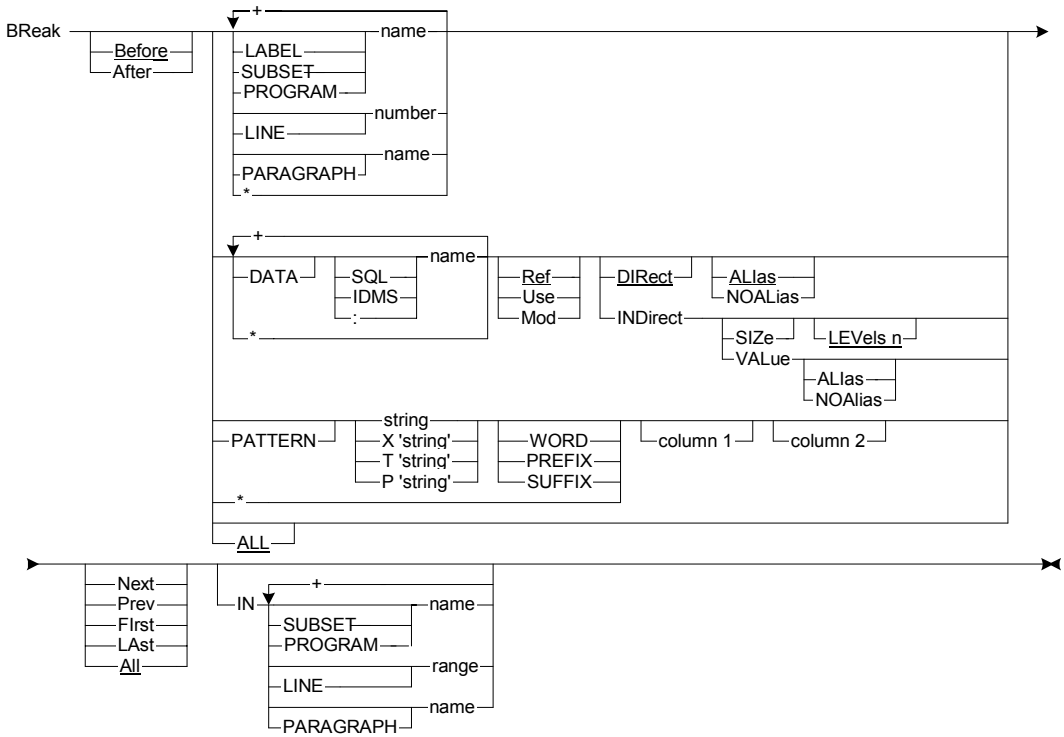
BRANCH

Positions the cursor at the specified target. BRANCH can be used to scroll from a statement such as a PERFORM, to the paragraph being performed. The BACKUP operand can then be used to return to the statement from which the branch occurred.

Bold operands are available only with ASG-Insight

BREAK

Inserts a Breakpoint before or after the statement containing the specified target. A Breakpoint creates an interrupt in the program execution. Program execution stops when the BREAK statement is encountered. The current status box displays with a STOPPED AFTER BREAK status. Typically, the BREAK command is used to automatically set numerous Breakpoints at various locations within the program, such as BREAK BEFORE CALLS ALL. Variables can be checked, values can be changed, or pseudo code can be entered when an interrupt is encountered during a SmartTest session.

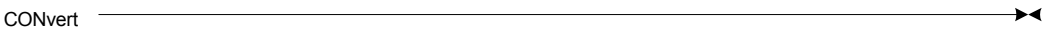


CANCEL

Terminates the current test session. When this command is entered on the Program View screen, the program being executed remains displayed but testing ends. The RUN or STEP command can be used to restart the test if desired. All pseudo code statements remain intact and Breakpoints remain active. For environments other than CICS, all files used by the program are closed and memory is freed. For CICS, all resources associated with the current task are released. When the CANCEL command is entered while connected to a batch job, the batch job step is terminated. Succeeding steps, if any, will then execute.

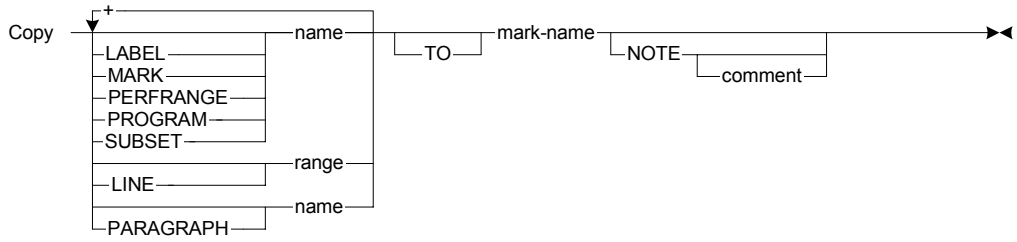
CONVERT

The CONVERT command displays the Convert Batch JCL screen.



COPY

Copies the contents of a path or set of lines to a mark. This provides a means of saving the same information under a different name for additional use. A new description can be included if desired. The same path or set of lines can be copied to different names. The specified TO *mark-name* must be unique. An error message displays and the copy function is not performed if an existing *mark-name* is specified.



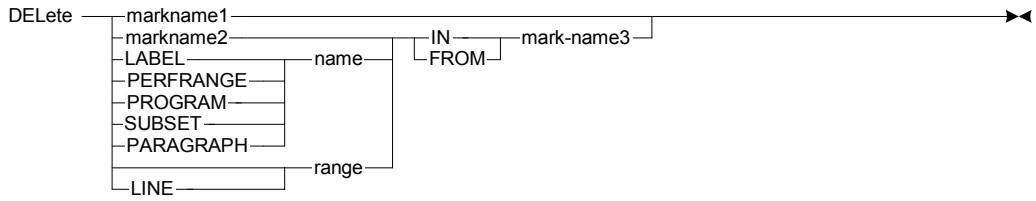
Available only with ASG-Insight

CURRENT

Saves the current location of the cursor for later use with the LOCATE &CURRENT command. When you use this command, your cursor must be positioned in the program source area.

DELETE

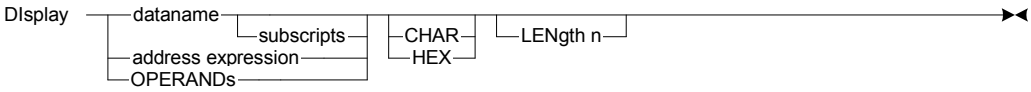
Deletes a mark name and its contents. If the mark name does not exist, an error message displays. The NETWORK and TRACK system-generated paths can also be deleted. SUBNETs are automatically deleted when the corresponding NETWORK is deleted. It is also used to delete a mark name, path, or set of lines from an existing mark name.



Available only with ASG-Insight

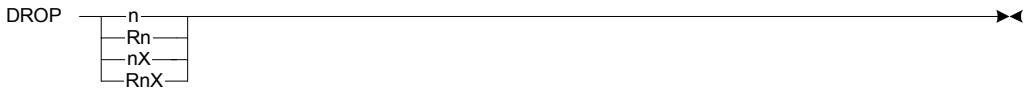
DISPLAY

Displays the current value of a specified data item in the long message area. The DISPLAY command supports pseudo code-defined variables. It does not support the pseudo code variable &COUNT.



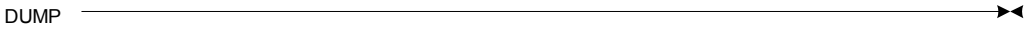
DROP

Ends addressability to any Assembler DSECT currently being addressed by Register *n*, as set by the USING command. The X is not required, even if the associated USING command had specified it. See "USING" on page 48 for more information. Full-screen Assembler support is available only if the SmartTest-ASM option is installed.



DUMP

Generates an MVS symptom and snap dump, or a CICS transaction dump of the current transaction being tested.



END

Terminates the current screen function and redisplays the previous screen. When the END command is entered on the primary screen, SmartTest is exited and control is returned to ISPF. SAVE options specified on the Options - Product Parameters pop-up determine if pseudo code, marks and equates are automatically saved in the AKR when exiting SmartTest.

ENVIRONMENT

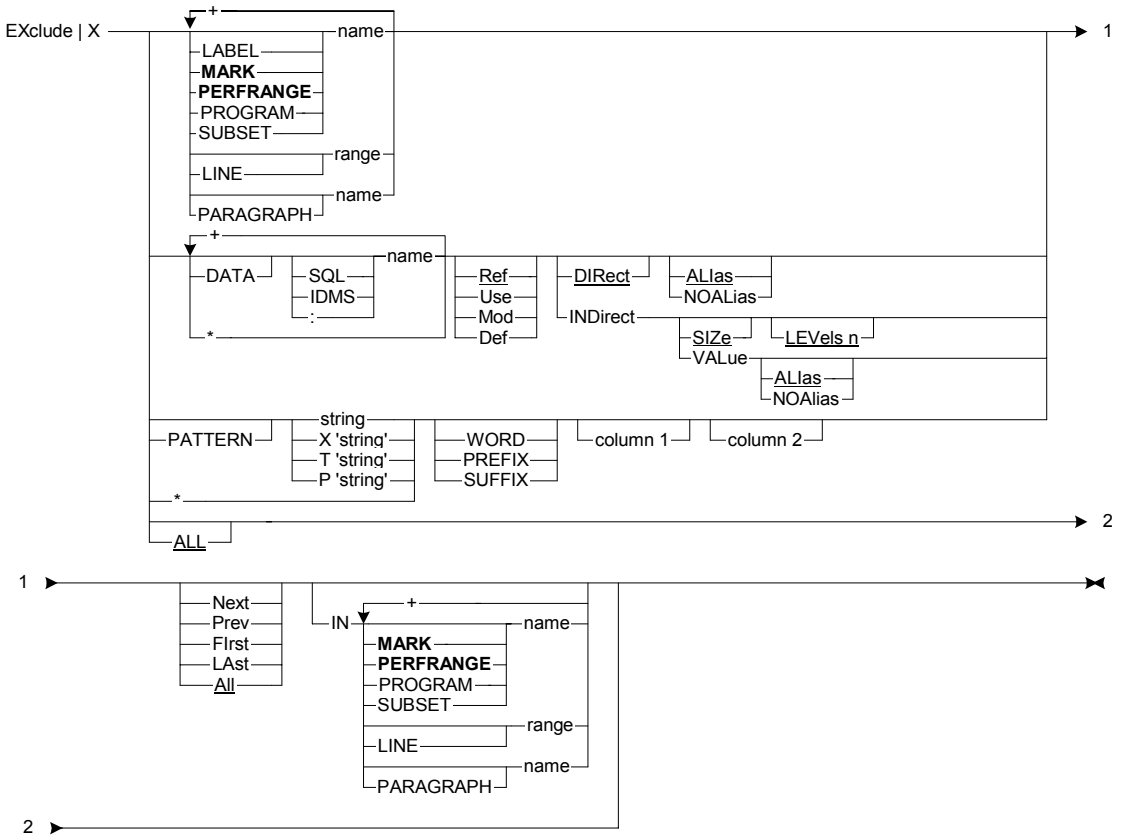
Displays the Environment Selection pop-up, used to select the SmartTest testing environment, specify the AKR to be used, and to specify the application load libraries and procedure libraries used for the non-CICS test session. This command can be entered on any SmartTest screen.

EQUATE

Defines a name for a character string. A character string can be a long command, operand set, pattern, dataname, etc. The equated names can be used during SmartTest sessions to reduce the number of keystrokes. Multiple equate statements can be used. To delete an equated name, type `EQUATE` and the name without the string. The substitution string can be changed on the List - Equates pop-up by typing over it with the new value.

EXCLUDE

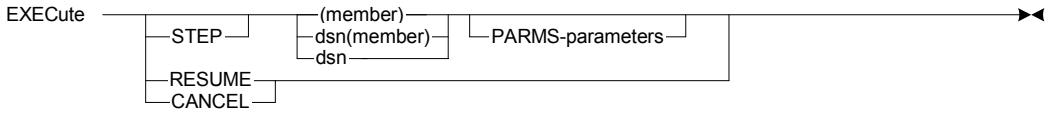
Performs a FINDXTND command on a specific target, excluding the resulting lines. Excluded lines are represented by a line of dashes and text stating *n* `LINE(S) NOT DISPLAYED`. The X or XX line commands can also be used to exclude lines from the screen.



`Bold operands are available only with ASG-Insight`

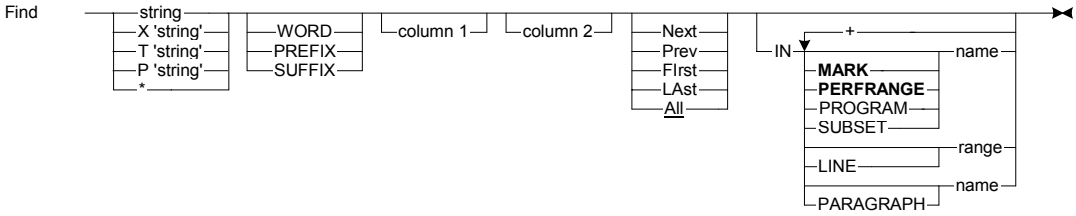
EXECUTE

Reads and executes a SmartTest script file. Script files can contain EXECUTE commands that execute lower level (nested) script files. The STEP operand causes each successive command in the script to be displayed in the command input area. The command can be changed if desired. When in STEP mode, typing `RESUME` executes the remaining script file commands without stepping through each. Typing `CANCEL` stops without processing the remaining script file commands. Script files that create a loop are recognized and an error message displays.



FIND

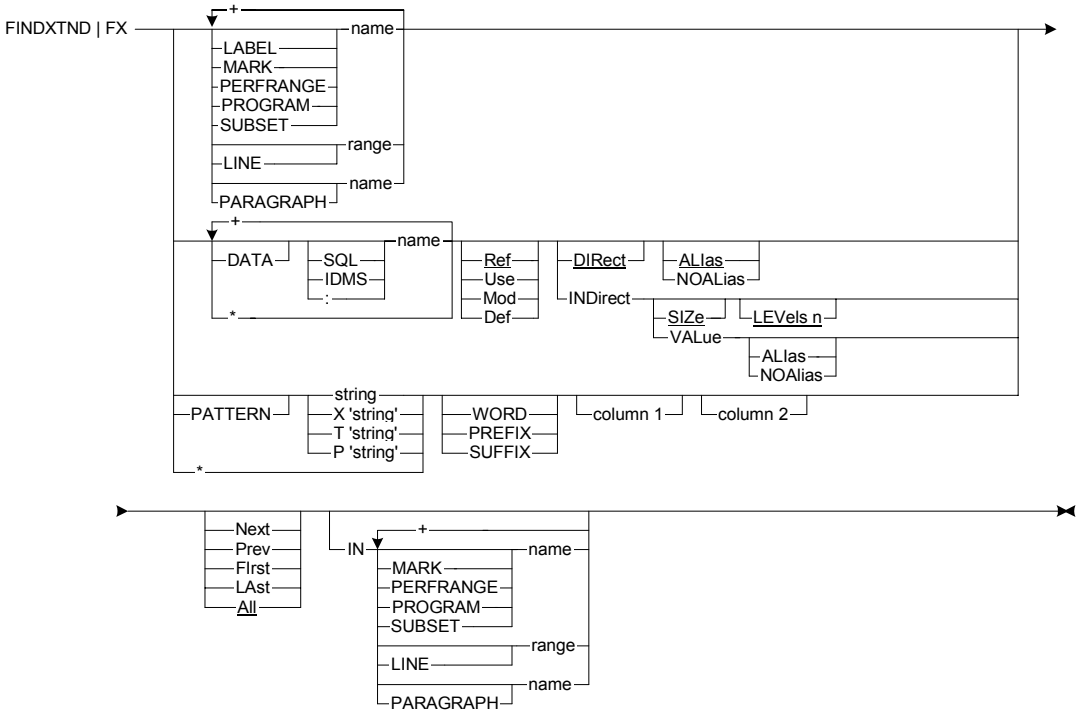
Searches for one or all occurrences of the specified character string. The syntax is similar to the ISPF/PDF FIND command. The search begins from the current line when the NEXT or PREV operands are specified. All lines are searched regardless of the current line or direction when the ALL operand is specified. Any excluded lines containing FIND targets are included and displayed. The IN operand restricts the FIND command to only the specified targets.



Bold operands are available only with ASG-Insight

FINDXTND

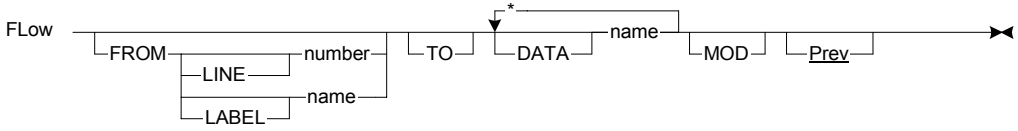
Performs a COBOL intelligent search of the source code for one or all occurrences of the specified target. Highlighting is used to indicate the occurrences found. If lines containing results are excluded, they are redisplayed on the screen. Tags indicating the type of target found are placed on the source code lines in columns 73 through 80.



Bold operands are available only with ASG-Insight

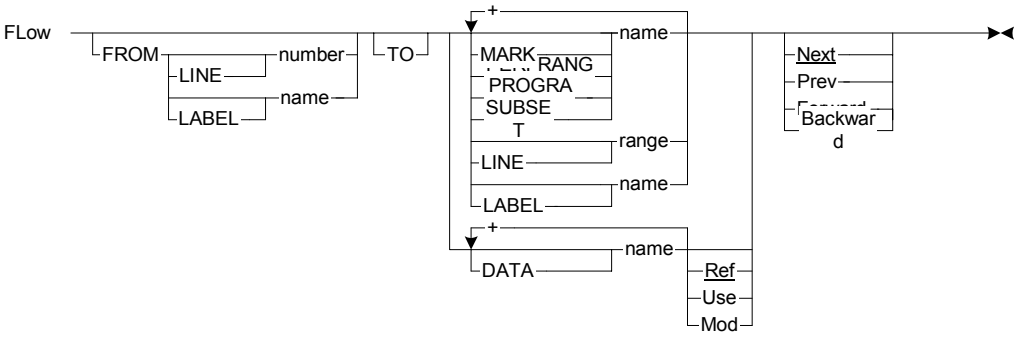
FLOW

Identifies all possible execution paths from a given point in a program to the specified target(s). Resulting paths are stored as NETWORK and SUBNETs. The FLOW command determines the execution flow of a program and indicates whether the specified targets can be reached from a given point. If the FROM operand is not specified, the cursor location is used as the starting point.



FLOW

Identifies all possible execution paths from a given point in a program to the specified target(s). Resulting paths are stored as NETWORK and SUBNETs. The FLOW command determines the execution flow of a program and indicates whether the specified targets can be reached from a given point. If the FROM operand is not specified, the cursor location is used as the starting point.



`Available only with ASG-Insight`

FORCE (CICS Only)

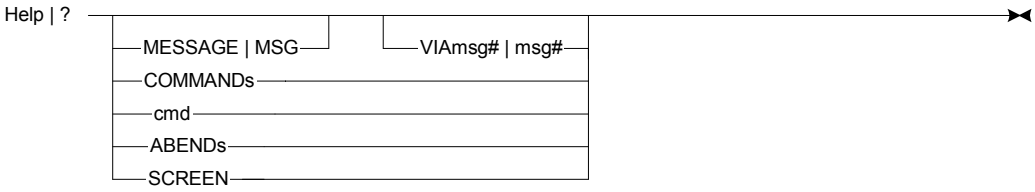
Forces an update of the Memory Display screen to be made after a storage-protected error message displays. You must have proper authorization to use this command. If FORCE is used to override SmartTest Storage Protection, take care to ensure that CICS is not corrupted during this override process.

GO

Transfers control to the statement containing the specified COBOL or Assembler label, PL/I label or procedure, pseudo code label, or line.

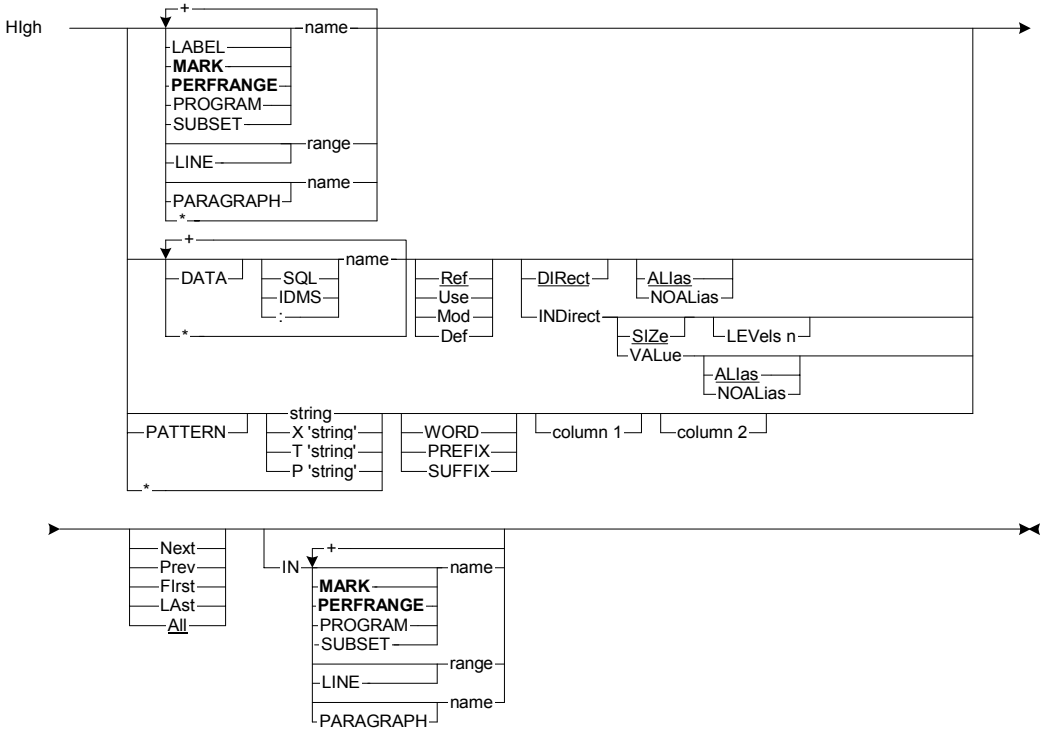
HELP

Displays information about the current SmartTest screen, command, or error message. Help is presented as a long message or a tutorial screen. After accessing the Help Tutorial, type `INDEX` and press Enter to display the Index, or type `TOC` and press Enter to display the Table of Contents.



HIGH

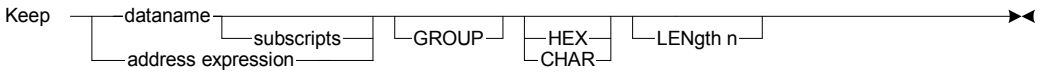
Highlights source code lines containing the specified targets. Lines already highlighted from prior commands are not reset.



Bold operands are available only with ASG-Insight

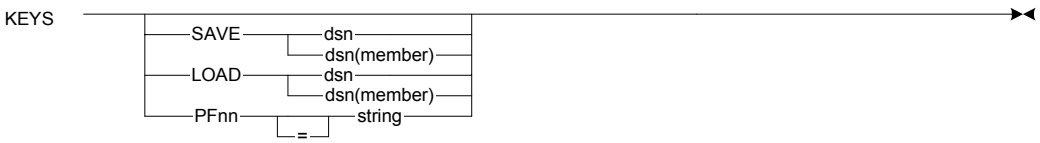
KEEP

Keeps the value and address of the specified data item displayed at the top of the Program View screen. The screen can be scrolled, and the kept lines remain displayed at the top of the screen. The number of lines kept is specified by the SET KEEP command. The maximum number of lines that can be specified in the Keep line area is 60. Lines can be deleted from the Keep line area using the D (Delete) line command.



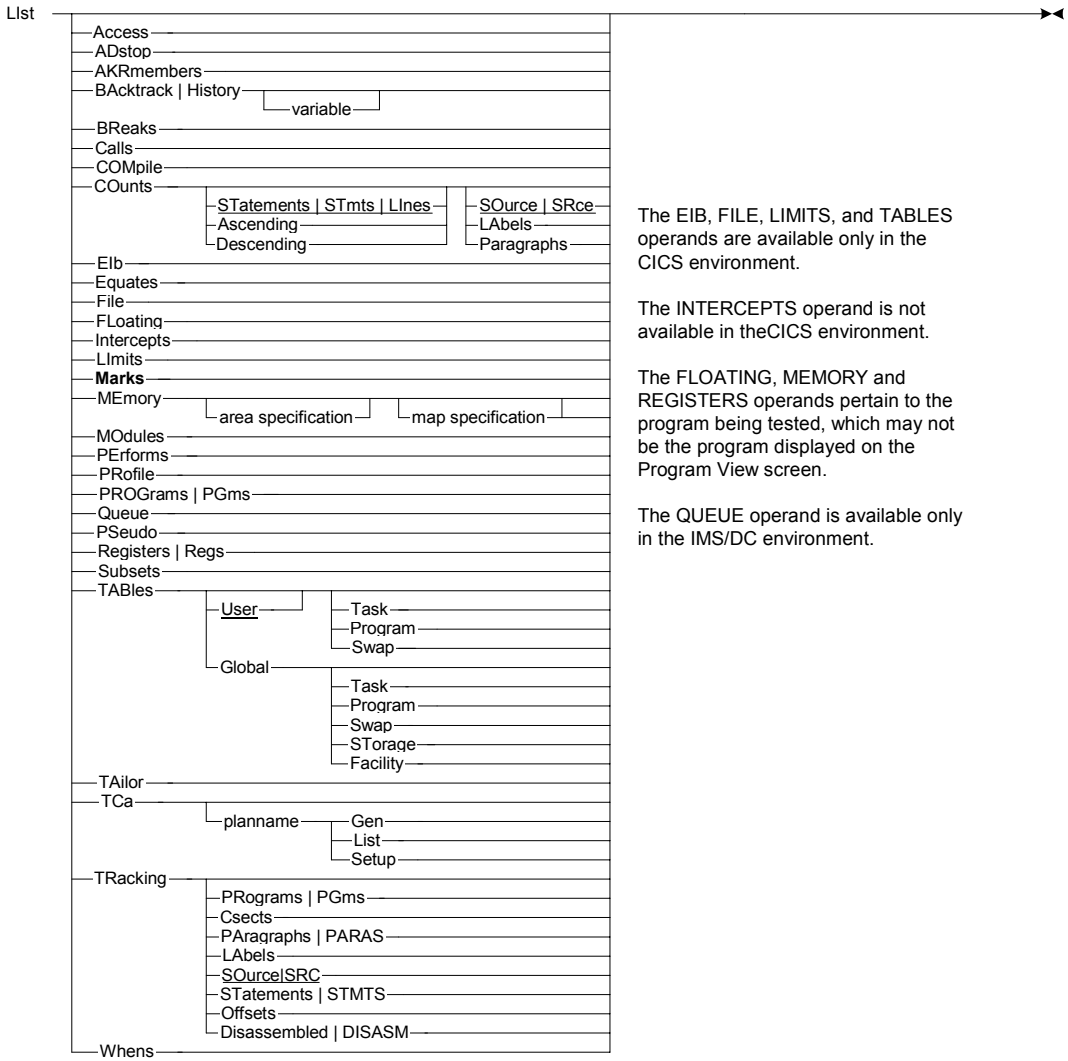
KEYS

Displays the Options - PF Key Definition pop-up. This screen is used to display and/or modify the current SmartTest PF key assignments. Values assigned to the SmartTest PF keys have no effect on other ISPF applications.



LIST

Displays the specified list screen.



The EIB, FILE, LIMITS, and TABLES operands are available only in the CICS environment.

The INTERCEPTS operand is not available in theCICS environment.

The FLOATING, MEMORY and REGISTERS operands pertain to the program being tested, which may not be the program displayed on the Program View screen.

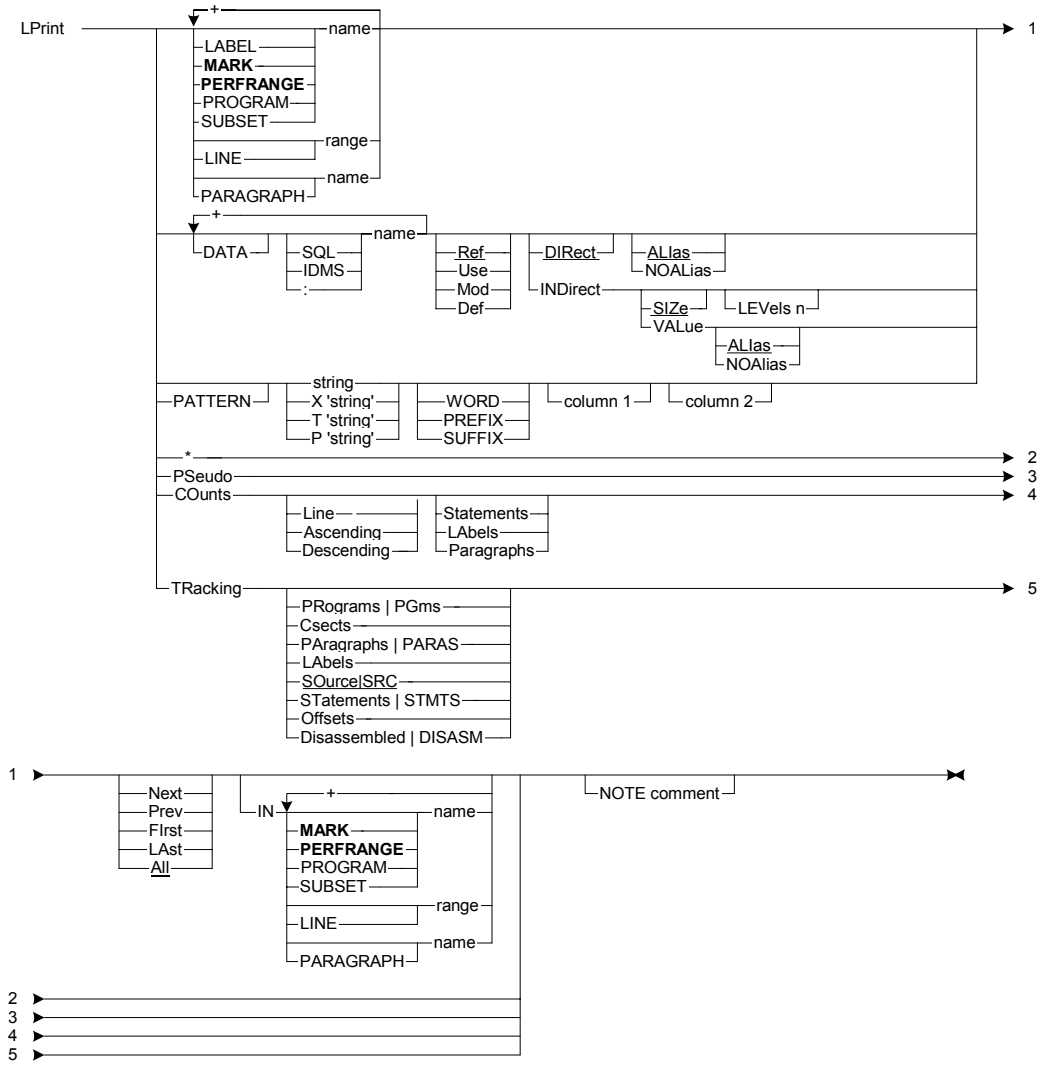
The QUEUE operand is available only in the IMS/DC environment.

LOCATE

On the Program View screen, this command displays the PROCEDURE DIVISION statement, the beginning of WHEN command statements, a specific line or label, or a hexadecimal offset within the program. From any directory or list screen, type `LOCATE` with a string to display an item that matches the specified string. Type `LOCATE *` to display the statement to be executed next.

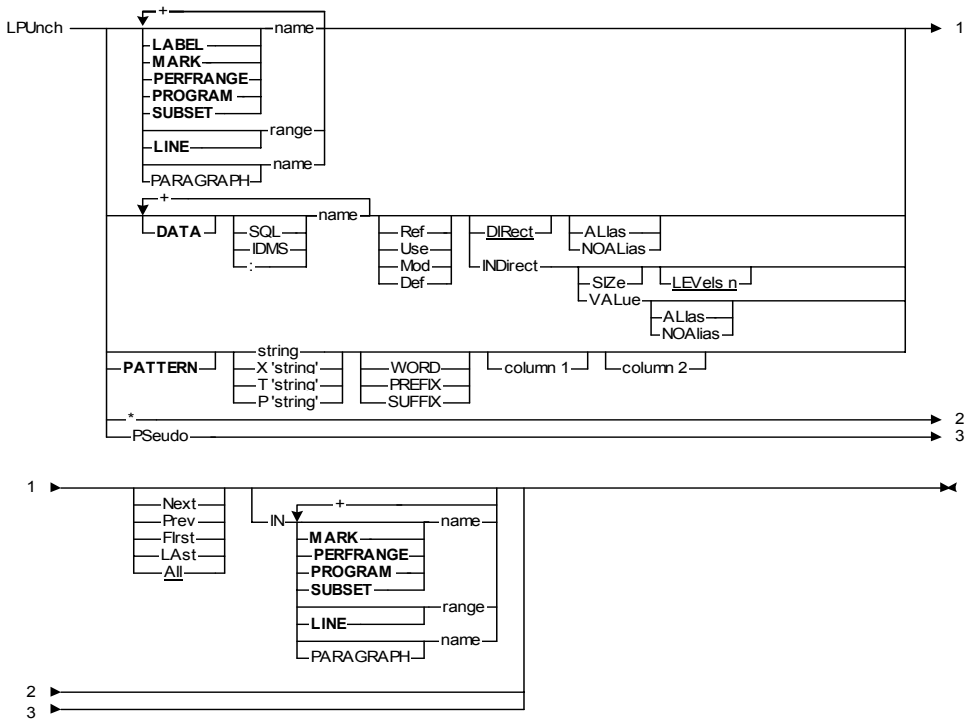
LPRINT

Copies lines containing the requested target to the List file.



LPUNCH

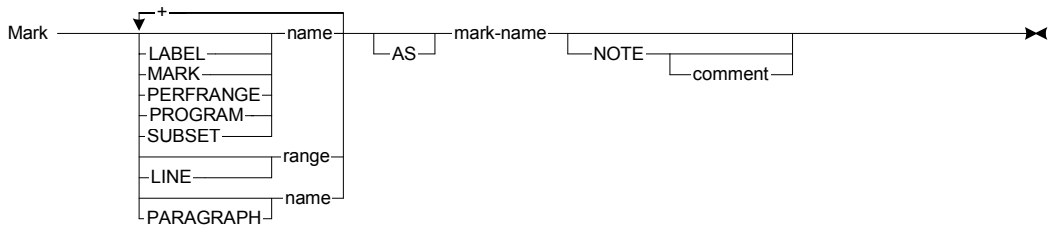
Copies lines containing the specified target to the Punch file for subsequent processing. Typing `LPUNCH *` causes the entire virtual screen (all data that can be viewed by scrolling down and up) to be copied to the Punch file. If Insight is not installed, only the operands * and PSEUDO are available.



Bold operands are available only with ASG-Insight

MARK

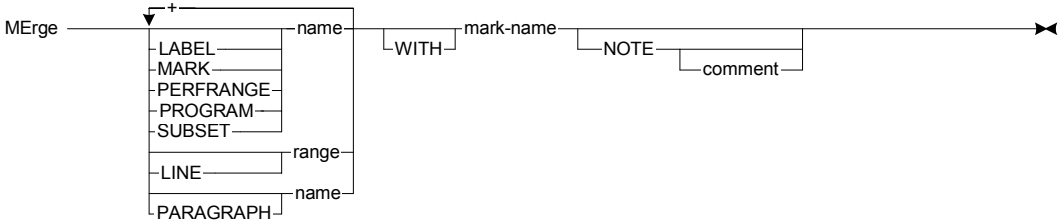
Saves the requested target as a mark set or path. An optional description can be included if desired.



Available only with ASG-Insight

MERGE

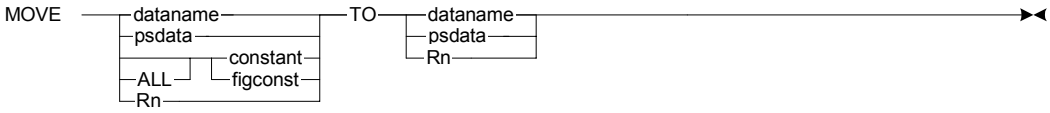
Adds the lines from the specified target to the specified mark name. If the specified mark name does not exist, it is created.



Available only with ASG-Insight

MOVE

Assigns the value contained in or represented by the first operand to the specified data item. The value is converted to the proper format for the data item if possible. If the value cannot be converted to the proper format, an error message displays.

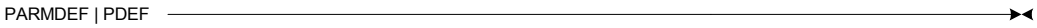


NEWCOPY (CICS Only)

Submits a CEMT NEWCOPY command for the specified module name to the current CICS region (the ACTIVE or CURRENT SYSID).

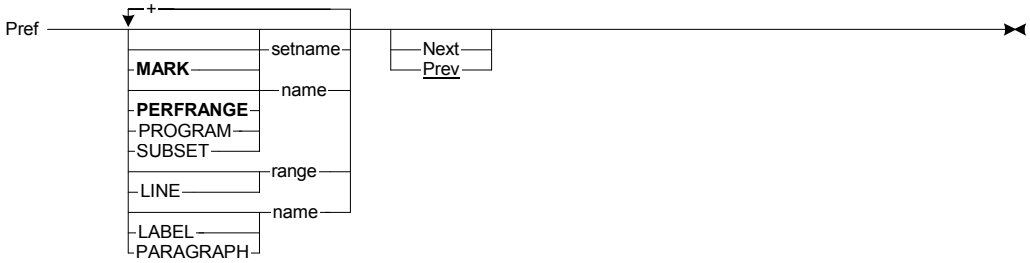
PARMDEF

Displays the Options - Product Parameters pop-up used to set parameters that affect the online operation of SmartTest.



PREF

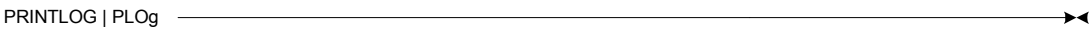
Displays the View - Paragraph Cross Reference pop-up for the requested target. The View - Paragraph Cross Reference pop-up shows how control is transferred to or from the target paragraphs.



Bold operands are available only with ASG-Insight

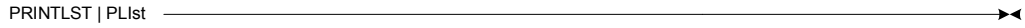
PRINTLOG

Displays the Options - Log/List/Punch Definition pop-up for processing of the Log, List, and Punch files.



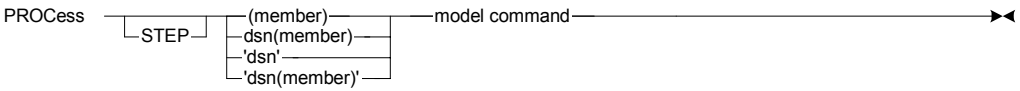
PRINTLST

Displays the Options - Log/List/Punch Definition pop-up for processing of the Log, List, and Punch files.



PROCESS

Defines an action to be performed repetitively on a file of input items. This file may contain the results list from an ESW product, or the file can be nested by the user. The model command can be any ESW product command that is valid in a script file and can be defined using substitution variables.



PRODLVL

Displays the current SmartTest and Center product level on the message line, including the product name, operating system, product release number, and level.

QUALIFY

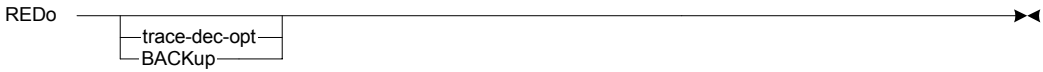
Displays a different program on the Program View screen. The program being tested remains active; that is, if a STEP or RUN command is entered, it is executed for the active program (the program being tested), NOT the qualified program. All other SmartTest commands are performed for the qualified program.

RECALL

Displays the previous ESW primary or internal command or message. The last 20 commands that have been executed and the last 20 messages that have been displayed are stacked. These commands or messages can be redisplayed in sequential order by typing `RECALL`. The POPUP operand redisplays the most recent pop-up.

REDO

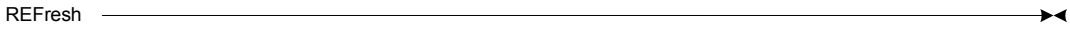
Executes the corresponding repeat command after execution of the FIND, FINDXTND, HIGH, PREF, SCROLL, or TRACE command. The REDO command operands are valid only if the last command was TRACE.



Bold operands are available only with ASG-Insight

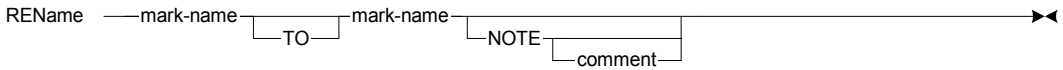
REFRESH

Brings in fresh versions of program summaries or copy members. This command is only used when one or more of the programs CALLed by the current program is analyzed while the current program is being viewed.



RENAME

Changes the name of a mark path or set of lines. An error message displays if the mark name specified to be renamed does not exist, or if the mark name specified to be assigned already exists.



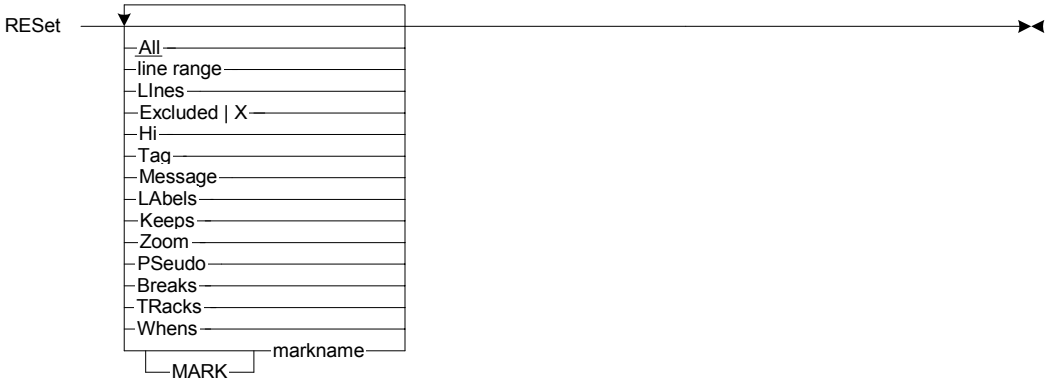
Available only with ASG-Insight

REPEAT

Re-executes the last stacked primary command from the cursor position.

RESET

Turns off highlighting, erases tags, redisplays excluded lines, cancels pending line commands, terminates message line displays, erases line labels, and deletes pseudo code statements, lines created as a result of one of the SmartTest Zoom line commands (ZD, ZA, etc.), WHEN statements, kept lines, or Breakpoints. All ISPF reset options are supported.



RETURN

Terminates the current screen function and returns to the primary screen. RETURN simulates multiple END commands, returning directly to the primary screen.

RFIND

Repeats the last FIND or FINDXTND command from the cursor position. The search is in the direction indicated in the last FIND or FINDXTND command.

RHIGH

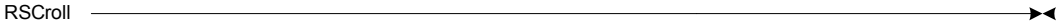
Repeats the last HIGH command from the cursor position.

RPREF

When on the Program View screen, redisplays the last View - Paragraph Cross Reference pop-up. When on the View - Paragraph Cross Reference pop-up, redisplays the last Program View screen.

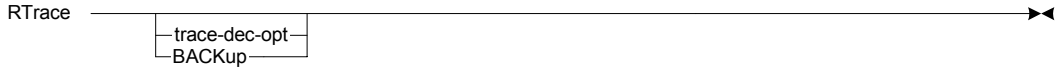
RSCROLL

Repeats the last SCROLL command from the cursor position.



RTRACE

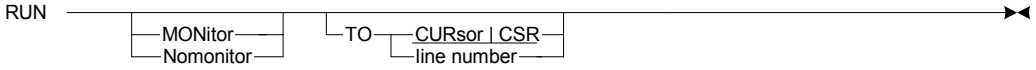
Continues the last TRACE command from where it stopped. Use BACKUP to return to the last decision point to follow a different path. Type `RTRACE` with no operand to display the Trace Decision Options pop-up.



`Available only with ASG-Insight`

RUN (Environments Other Than CICS, BACKTRACK OFF)

Begins testing of a program or resumes testing after an interrupt. Interrupts can result from Breakpoints or error conditions. The SET MONITOR command determines the default operand.

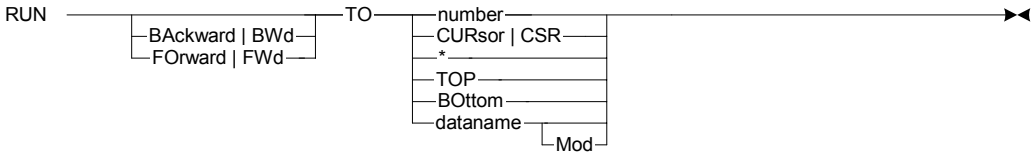


RUN (CICS Environment, BACKTRACK OFF)

Initiates a transaction or resumes execution of an active test session after an interrupt occurs. Interrupts can result from a Breakpoint or an error condition. If there are no interrupts, the program executes through completion. The STEP command can be used instead of the RUN command to execute the program statement by statement if Single Step is set to Yes on the Session Tailoring screen. The RUN command can also be used to toggle from ISPF to the current CICS region.

RUN (BACKTRACK ON)

Begins backward or forward testing of a program or resumes backward or forward testing after an interrupt. Interrupts can result from Breakpoints or error conditions. The direction is determined by the last command to specify BWD or FWD.



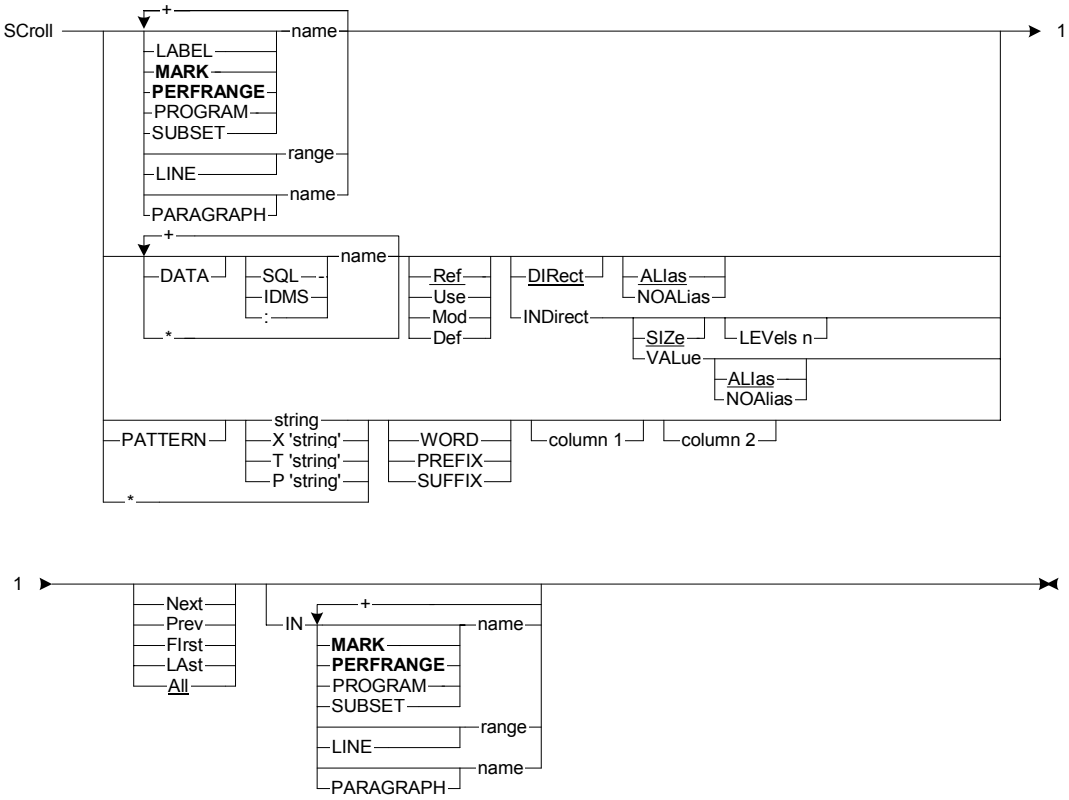
SAVE

Saves pseudo code, marks, and/or equates in the AKR. When you type `SAVE` with no operands, the Save Options pop-up displays.


```
Bold operands are available only with ASG-Insight
```

SCROLL

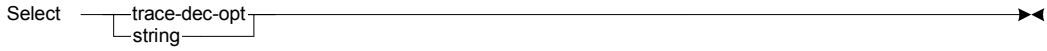
Scrolls to the first line containing the specified target. Highlighted lines remain unchanged.



Bold operands are available only with ASG-Insight

SELECT

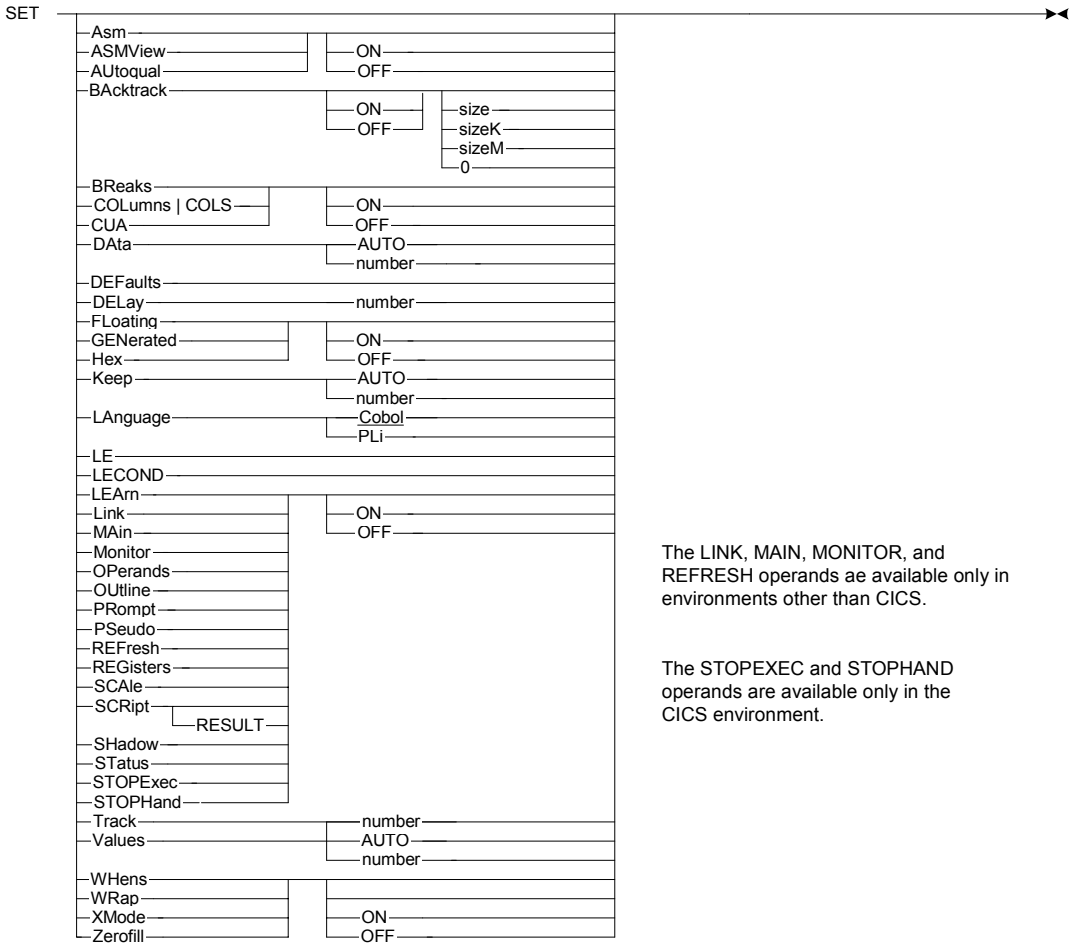
Selects an option on the Trace Decision Options pop-up. Type `SELECT` followed by the desired option in the command input area. An S can be entered on any screen with a selection field to the left of the displayed items to select a particular item.




```
Bold operands are available only with ASG-Insight
```

SET

Enables or disables the mode indicated by the specified operand. Entering the SET command with only a mode operand functions as a toggle switch (e.g., SET ASM sets the mode to ON if the current value is OFF).

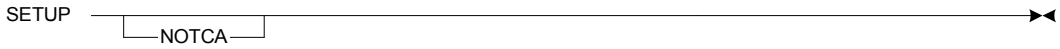


The LINK, MAIN, MONITOR, and REFRESH operands ae available only in environments other than CICS.

The STOPEXEC and STOPHAND operands are available only in the CICS environment.

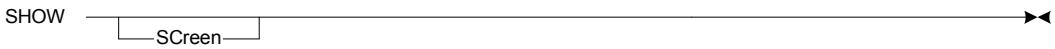
SETUP

Displays the Session Setup screen for the currently selected environment.



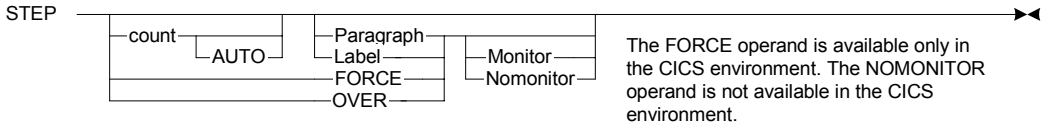
SHOW (CICS Only)

Redisplays the last user application screen. This command is only available during an active test session.



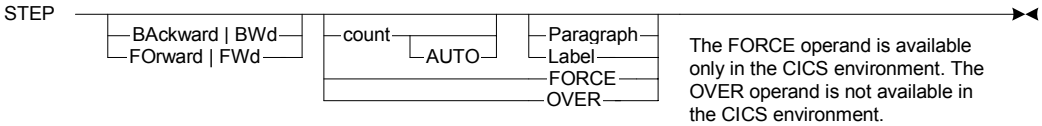
STEP (BACKTRACK OFF)

Steps through a program, resumes testing after an interrupt occurs, or continues stepping through program execution after a STEP command has been entered. Interrupts can result from a Breakpoint or an error condition. Use the SET command to set the increment for the STEP command to the COBOL statement level, the Assembler instruction level, or the PL/I statement level. If a storage violation occurs in the CICS environment, do not use the FORCE operand to continue execution without understanding the significance of the storage violation warning.



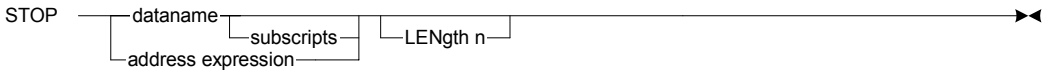
STEP (BACKTRACK ON)

Enters, exits, and simulates execution backward and forward in the statement execution history. The Backtrack Recording facility must be active for this command to be valid.



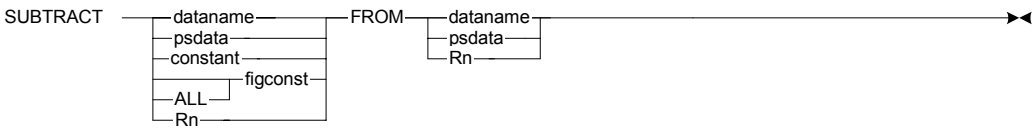
STOP

Sets an address stop for a specified data item. Address stops are inactive when you are not running SmartTest MONITORed.



SUBTRACT

Subtracts the value contained in or represented by the first operand from the specified data item. The value is converted to the proper format for the data item.

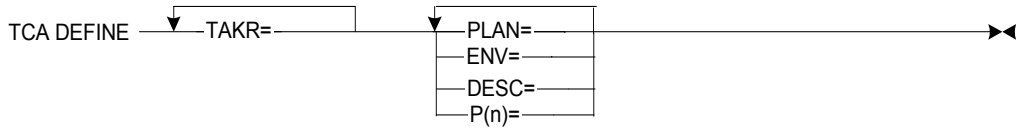


TCA Super Commands

Super commands are command phrases with parameters that enable you to perform some of the same functions represented by standard TCA commands using fewer lines of text. You can enter a single super command on multiple lines since continuation occurs when a command line ends in a comma. Parameters cannot span lines.

TCA DEFINE

Specifies an existing AKR to be used as the TCA AKR and to define a plan in a defined TCA-specific AKR.



TCA LIST

Opens a TCA plan. You can also export or delete the results in the plan.

TCA RECORD

Records the execution coverage information into a TCA plan.

TCA REPORT

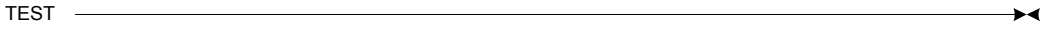
Generates reports from the results in a TCA plan.

TCA RUN

Executes a test session using the information stored in the TCA plan.

TEST

Displays the Test/Program View Entry screen.



TESTPOINT

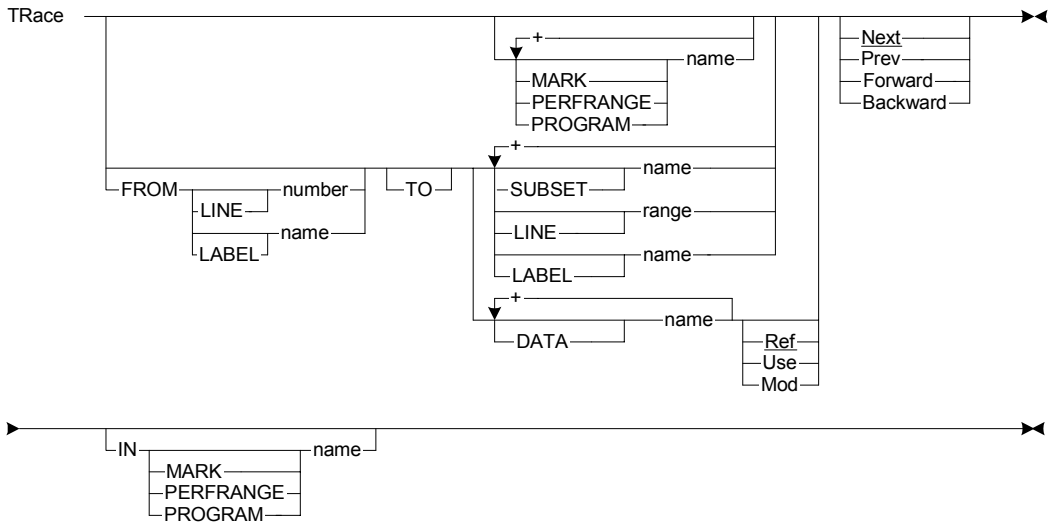
Displays the Test - SmartTest Testpoint Generation screen, which is used to specify necessary criteria when setting breakpoints with an impact dataset.

TOGGLE

Toggles from the SmartTest ISPF environment to the connected CICS or IMS/DC environment. The TOGGLE command can also be entered on the CICS screen to return to the ISPF environment. The IMS/DC/TOGGLE command can be entered on the IMS/DC screen to return to the ISPF environment.

TRACE

Follows the execution of a program, searching for the specified target. The MARK and PERFRANGE operands trace the path represented by the MARK or PERFRANGE name. Tracing to a SUBSET name, LINE range, LABEL name, or DATA name traces from the starting point to the lines represented by these targets.



Available only with ASG-Insight

UPDATE

Changes pseudo code lines to actual COBOL source lines, making them part of the program. This command can only be entered while in the EDIT facility.

USING

Specifies which Register to use as a base for determining the address of fields within Assembler DSECTs. The ZOOMDATA command uses this information to display data fields that are DSECT relative. Full-screen Assembler support is available only if the SmartTest-ASM option is installed.

UTILITY

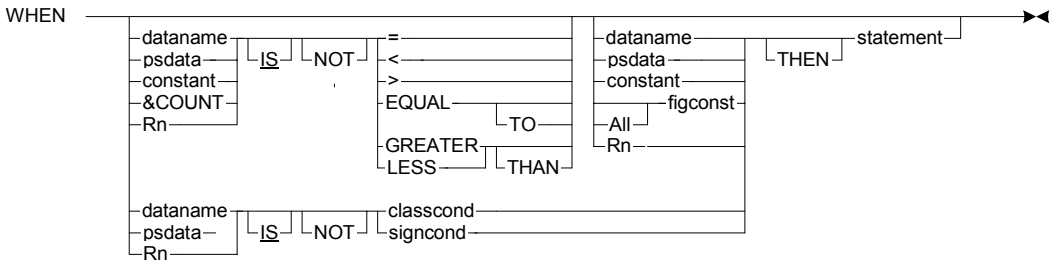
Displays the File - AKR Utility pop-up, used to display the member directory, allocate or expand an AKR, and rename or delete a member.

VIEW

Displays the Program View screen for the current qualified program.

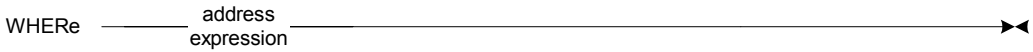
WHEN

Inserts a WHEN pseudo code statement into the source program immediately preceding the bottom-of-data line. WHEN evaluates the conditional expression and if true, the imperative statement is executed. Testing of conditional expressions is performed the same as in COBOL conditional expressions. Typing `WHEN` with no operands places the user in input mode. Because the WHEN pseudo statement causes every instruction in the program to be tested for the specified condition, this statement can have a significant performance impact.



WHERE

Identifies a storage location. A message displays that indicates the area of the program where the address is located and the offset.

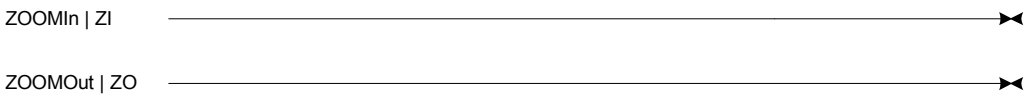


ZOOMDATA

Scrolls to the definition of the specified dataname, and displays the value and address of the dataname.

ZOOMIN/ZOOMOUT

Displays or excludes source code lines according to the hierarchical levels of the program. Use in conjunction with each other to show the structure of a program and provide a means of stepping through each level or going directly into or out of a particular section of source code.

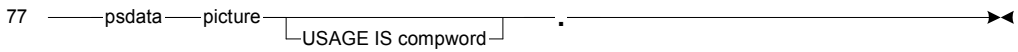


2 Pseudo Code Statements

Pseudo code statements are entered in line with existing source code, and are used to insert temporary COBOL code during a test session.

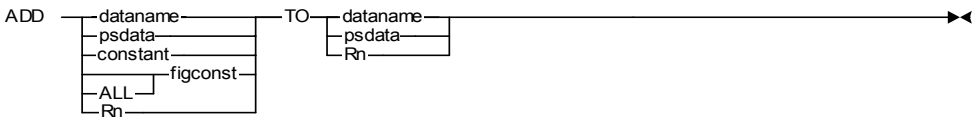
77 (Pseudo Code Data Item)

Defines a pseudo code data item. Data items are entered in Area A (columns 8 through 11) within a block of pseudo code. Each data item must be unique and cannot be qualified. If entered in the PROCEDURE DIVISION, data items must be manually moved to the DATA DIVISION when pseudo code is updated with source code.



ADD

Adds the value contained in or represented by the first operand to the specified data item. The value is converted to the proper format for the data item.

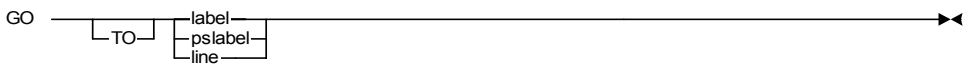


BREAK

Forces a Breakpoint before a specific statement. A pseudo code BREAK statement causes an unconditional interrupt in the program execution.

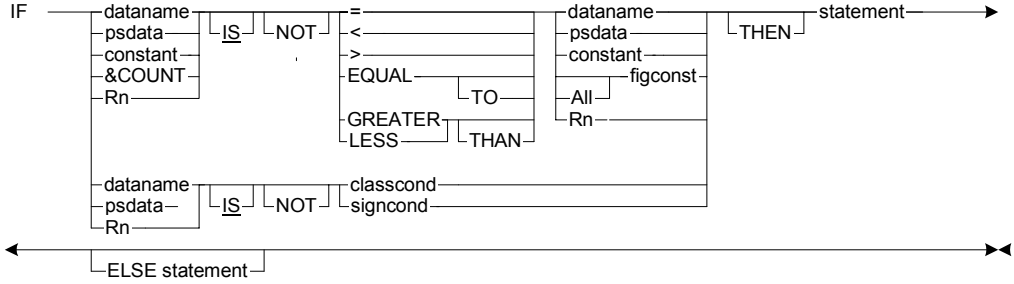
GO

Transfers control to the statement containing the specified COBOL or Assembler label, pseudo code label, or line.



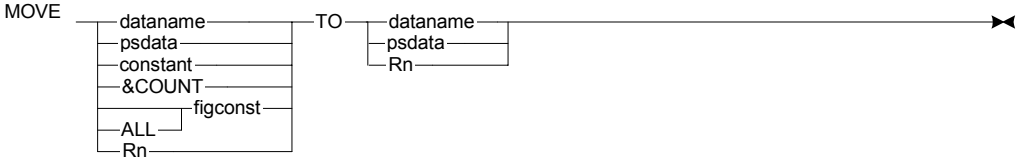
IF

Tests conditional expressions and if the condition is true, the imperative THEN clause is executed. If the condition is false, the imperative statement following the ELSE clause or the next sentence is executed.



MOVE

Assigns the value contained in or represented by the first operand to the specified data item. The value is converted to the proper format for the data item if possible. If the value cannot be converted to the proper format, program execution stops and an error message displays.

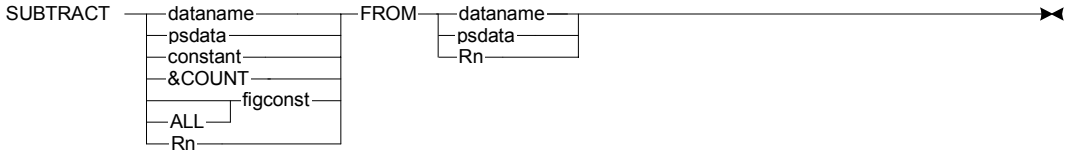


Pslabel. (Pseudo Code Label)

Defines a pseudo code paragraph name. The name is 1 to 30 alphanumeric characters, beginning with an alphabetic character, that is entered in Area A (columns 8 through 11) and is referenced by a GO statement. The name cannot be an existing data or label name.

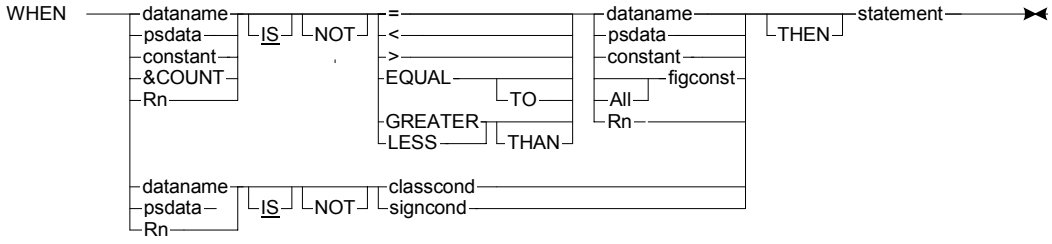
SUBTRACT

Subtracts the value contained in or represented by the first operand from the specified data item. The value is converted to the proper format for the data item.



WHEN

Evaluates the conditional expression and if true, the imperative statement is executed. Testing of conditional expressions is performed the same as in COBOL conditional expressions.



3 Operand Definitions

Operand	Description
*	Reuses the target of the previous command. This can be concatenated once with any number of other operands; however, the * operand cannot be concatenated to itself.
: *name*	Includes datanames that are COBOL variables only.
address	An absolute memory address.
address expression	An address expression consists of either an address or a register number followed by a maximum of 32 indirection indicators (e.g., R14?).
classcond	The COBOL reserved words used to test class conditions (e.g., ALPHABETIC, NUMERIC).
cmd	A SmartTest primary command.
Column1	The column number where the search is to begin.
Column2	The column number where the search is to end.
Comment	Text following the NOTE operand that describes the name.
compword	The standard COBOL computational reserved words used to describe data items used in arithmetic operations (e.g., COMPUTATIONAL, COMP-3).
constant	A numeric or non-numeric literal.

Operand	Description	
Count	A specified number of COBOL statements or Assembler instructions.	
DATA *name*	A COBOL dataname or qualified COBOL dataname. DATA name refers to any valid COBOL reference for a data element. The DATA name operand also accepts:	
	Use	Occurrences of the dataname where its value is tested or used (including aliases unless using NOALIAS).
	Mod	Occurrences of the dataname where its value is set or modified (including aliases unless using NOALIAS).
	Def	Definitions for the dataname in the DATA DIVISION (including aliases unless using NOALIAS).
	Ref	Use, Mod and Def.
	ALIas	Includes aliases of the dataname.
	NOAlias	Does not include aliases.
	DIRect	Considers only the specified dataname.
	INDirect	Includes occurrences of any dataname indirectly affected by the specified dataname (and aliases, if specified).
	SIZe	Considers datanames indirectly affected by a change in the size of the specified dataname.
	VALue	Considers datanames directly or indirectly affected by a change in the value of the specified dataname.
	LEVels nbr	Identifies the depth of the indirect references.
dataname	A data item for which the value and address is to be displayed.	

Operand	Description
dsectname	An Assembler DSECT name.
dsn	A sequential dataset (script file) that contains user-specified commands to be executed. This dataset must be in card image format (LRECL=80), and entered in the format: sequential.dataset.name
dsn(*member*)	A partitioned dataset member (script file) that contains user-specified commands to be executed. This dataset must be in card image format (LRECL=80), and entered in the following format: pds.dsn(member)
figconst	The COBOL reserved words (figurative constants) used to test specific values (e.g., SPACES, LOW-VALUES).
hexoffset	A hexadecimal offset within the program (e.g., X'0F14').
IDMS *name*	Includes datanames that are IDMS variables only.
label	A standard COBOL or Assembler label name referenced in a pseudo code statement.
.*label*	A line label entered in the prefix area (columns 1 through 6) on the Program View screen.
LABEL *name*	Any paragraph or section name of the PROCEDURE DIVISION, as well as the literals PROCEDURE and PROC. This includes all transfers of control to the label name.
line	A line number of a source code statement.
LINE *number*	A single line number.

Operand	Description
LINE *range*	A single line number or range of lines.
mark-name	A name assigned to a path or set of lines using the COPY, MARK, MERGE or RENAME command, or a system-generated path. Mark-name can be a maximum of 10 alphanumeric characters, beginning with an alphabetic character, and can include hyphens.
MARK *name*	A 1 to 10 alphanumeric character name given to a set of lines or a path using the COPY, MARK, MERGE or RENAME command, or one of the following system-generated paths: TRACK | TRK — Created by a TRACE command. NETWORK | NET — Created by a FLOW command. SUBNET*n* | SUB*n* — A path created by the FLOW command that reaches from the beginning of the NETWORK to one of the results.
MARK *setname*	A 1 to 10 character alphanumeric name given to a set of lines using the COPY, MARK, MERGE or RENAME command.
model command	Any ESW product command that is valid in a script file and can be defined using substitution variables.
msg#	The number of an error message or information message.
name	A name for a character string. Name can be a maximum of 10 alphanumeric characters, beginning with an alphabetic character, and can include hyphens. Name can also be a PERFRANGE name, LABEL name, DATA name or SUBSET name. Refer to the specific operand for more information.
NOTE comment	A user-supplied description.

Operand	Description
number	A single line number.
PARAGRAPH *name*	Any paragraph or section name of the PROCEDURE DIVISION, as well as the literals PROCEDURE and PROC. PARAGRAPH name includes the entire paragraph or section.
PATTERN *string*	A string of alphanumeric characters. If the string contains blanks, it must be enclosed in quotes. The PATTERN string operand also accepts: X 'string' Hexadecimal string. T 'string' Text string. P 'string' Picture string. WORD Pattern string directly preceded and followed by any non-alphanumeric characters other than hyphens. PREFIX Word that begins with the pattern. SUFFIX Word that ends with the pattern.
PERFRANGE *name*	The name specified in a PERFORM statement. It includes all the statements that are executed as a result of a PERFORM statement. The name of any section contained in the Declaratives.
picture	A standard COBOL picture clause that defines the type of data that a data item will contain.
pgm	A CSECT contained in a load module.
PROGRAM *name*	The name of the main program or any nested program representing all the code contained in the program. This includes all the programs physically nested inside the specified program.

Operand	Description
psdata	A level 77 data item defined within pseudo code. A pseudo code dataname.
pslabel	A label name defined within pseudo code.
range	A line or range of lines. Line numbers are displayed in columns 1 through 6.
R*n*	A register 0 through 15.
setname	A MARK name of type set (i.e., cannot be a path). Refer to the MARK setname operand for more information.
signcond	The COBOL reserved words used to test sign conditions (i.e., NEGATIVE, or POSITIVE).
SQL *name*	Includes datanames that are DB2/SQL variables only.
stmt	A pseudo code statement that is entered in line with existing source code.
string	A character string to be substituted by the EQUATE command. Also, a string of alphanumeric characters. Refer to the PATTERN string operand for more information.

Operand	Description
SUBSET *name*	COBOL verbs of a similar nature that have been grouped together. Predefined COBOL language subsets (**Bold** subsets are available only if an Extended Analysis has been performed):

ASsignment	DEBug	IO
CAll	DEFinition	LABel
CIcs	DIRective	MAINline
COBOLII	DIVision	MATH
COBOL/370	DL/I | DL/1	Output
COMment	DML	PARagraph
CONditional	ENtry	PERform
COPy	EXIt | PGMExit	RETurn
DB2/SQL	FALLthrough	SECtion
DDL	GOto	SORTMerge
DEAD	IDMS	STructure
DEADCode	INClude	TAG
DEADData	Input	TESted | UNTested

Note:

TESted and UNTested subsets are available only if you have applied TCA results. See the *ASG-SmartTest TCA User Guide* for more information.

Screen subsets:

Highlighted|HI	Excluded|X
NONHighlighted|NHI	NONExcluded|NX

A tagged lines subset of tags appearing in columns 73 through 80 of the Program View screen.

Operand	Description
trace-dec-opt	An option as listed on the Program View screen or an option listed on the Trace Decision Options pop-up.

4 Program View Line Commands

Line Command	Description
.label	Assigns a symbolic name to a source code line. The name must be a period followed by 1 to 5 alphabetic characters.
A	After. Indicates the line after which the results of an operation such as Copy or Move are to be placed.
B	Before. Indicates the line before which the results of an operation such as Copy or Move are to be placed.
BR	Break. Inserts a Breakpoint before a specific line. If inserted on a non-executable line, the Breakpoint is inserted before the next executable statement.
C*n*	Copy. Copies a pseudo code or source code line or group of lines to another location in the source code. COBOL source code statements that are copied become pseudo code lines. The default is 1.
CC	Copy Block. Copies a block of pseudo code or source code lines to another location in the source code.
D*n*	Delete. Deletes a pseudo code line or group of lines. If the value specified is greater than the number of available lines, all remaining lines are deleted. The default is 1.
DD	Delete Block. Deletes a block of pseudo code lines or a block of keep lines.

Line Command	Description
F*n*	First. Redisplays the specified number of excluded lines, starting with the first line in the block. The default is 1.
GO	Go. Makes the line on which the GO command is entered the next line to be executed. GO should only be used within the paragraph currently being executed. GO can only be used on source code lines that are executable.
H*n*	Highlight. Highlights a line or group of lines. If the value specified is greater than the number of available lines, all remaining lines are highlighted. The default is 1.
HH	Highlight Block.
I*n*	Insert. Inserts one or more blank lines into the source code.
K*n*	Keep. Keeps the display of the value and address of data items at the top of the screen. When the screen is scrolled, the kept lines remain displayed at the top of the screen.
KG*n*	Keep Group. Keeps the display of levels, values and addresses of group items at the top of the screen.
KGH*n*	Keep Group Hexadecimal. Displays the levels, values and addresses of group items in hexadecimal format.
KH*n*	Keep Hexadecimal. Keeps the display of the value and address of a data item in hexadecimal format at the top of the screen.
L*n*	Last. Redisplays the specified number of excluded lines, starting with the last line in the block. The default is 1.
M*n*	Move. Moves a pseudo code line or group of lines to another location in the source code. The default is 1.

Line Command	Description
MM	Move Block. Moves a block of pseudo code lines to another location in the source code.
R*n*	Repeat. Replicates a pseudo code or source code line a specified number of times. COBOL source code lines that are repeated become pseudo code lines. The default is 1.
RR	Repeat Block. Replicates a block of pseudo code or source lines. COBOL source lines become pseudo code lines.
S*n*	Show. Redisplays the specified number of excluded lines, starting with the first line in the block. The default is 1.
SS	Show Block. Redisplays a block of lines that were excluded.
X*n*	Exclude. Excludes a line or group of lines from being displayed. Excluded lines are replaced with a row of dashes. The default is 1.
XX	Exclude Block. Excludes a block of lines from being displayed.
ZA	Zoom Assembler. Displays Assembler instructions that correspond to a COBOL or PL/I source statement.
ZD*n*	Zoom Data. Displays the value and address of data items. The n operand is the relative position of the data item on the line to be displayed.
ZG*n*	Zoom Group. Displays the levels, values and addresses of group items. The n operand is the relative position of the data item on the line to be displayed.
ZGH*n*	Zoom Group Hexadecimal. Displays the levels, values and addresses of group items in hexadecimal format. The n operand is the relative position of the data item on the line to be displayed.

Line Command	Description
ZH*n*	Zoom Hexadecimal. Displays the value and address of a data item in hexadecimal format. The n operand is the relative position of the data item on the line to be displayed.
ZI	Zoom In. Performs the ZOOMIN primary command at the line on which it is entered.
ZO	Zoom Out. Performs the ZOOMOUT primary command at the line on which it is entered.

5 Analyze Options

Options		
BUF(*nnnnn*K) BUF=*nnnnn*K	IO(*x,x,...x*) IO=*x* NOIO	RETurn(*x,x,...x*) RETurn=*x* NORETurn(*x,x,...x*) NORETurn=*x*
COBOL370 COBOLII COB2R3 NOCOBOLII	LANGLVL(1|2) lineCNT=60	SEQ NOSEQ
DB2LIB=*xxxxxx.xxxxx.xxxxxx*	MAIN	SOUrce NOSOUrce
DB2PLAN=*xxxxxxxx*	MBRERCNT=*nnnn*	spACE1|2|3
DYNcall NODYNcall	Output(*x,x,...x*) Output=*x* NOOutput(*x,x,...x*) NOOutput=*x*	SQLID=*nnnnnnnn* SQLID(*nnnnnnnn,nnnnnnnn,nnnnnnnn*

Options		
fLAGW fLAGE fLAG(*x*)	PROgram(*xxxxxxxxxx*)	SUBSYS=*xxxx*
Input(*x,x,...x*) Input=*x* NOInput(*x,x,...x*) NOInput=*x*	RECur NORECur	XLIVE XMEM

6 VIASUB/VIASUBDS Parameters

Parameters		
AKR(*xxxxx*)	OUTPUT(*xxxxx*)	SD NOSD
AOPT(*xxxxx*)	PANEL NOPANEL	SDR NOSDR
CMPL NOCMPL	PGM(*xxxxx*) PROCONLY	SDX NOSDX
DSCHK NODSCHK	REUS NOREUS	ST NOST
EDIT	ENS NOENS	STX NOSTX
INS NOINS		

7 Assigning PF Keys

Primary Defaults

PF Keys		
PF1/13 HELP	PF2/14 SPLIT	PF3/15 END
PF4/16 RUN	PF5/17 RFIND	PF6/18 STEP
PF7/19 UP	PF8/20 DOWN	PF9/21 SWAP
PF10/22 BRANCH	PF11/23 BRANCH BACKUP	PF12/24 RECALL

Suggested Alternate PF Keys

PF Keys		
PF1/13 LIST MEMORY	PF2/14 SPLIT	PF3/15 END
PF4/16 RUN TO	PF5/17 REPEAT	PF6/18 STEP OVER
PF7/19 LIST	PF8/20 SET	PF9/21 SWAP
PF10/22 LEFT	PF11/23 RIGHT	PF12/24 RECALL MSG

8 Storage Area Keywords

CICS Keywords

Keyword	Description
AFCB	Displays the Authorized Function Control Block
AICB	Displays the CICS Application Interface Control Block
CIA	Terminal User Area (same as TUA)
COM	Command Level Common Storage Area
CSA	Common System Area
DCA	Dispatch Control Area
DCT-*name*	Destination Control Table entry - Name is the transient data ID
DWE-*nn*	nth Deferred Work Area
EIB	EXEC Interface Block
EIS	EXEC Interface Structure
FCT-*name*	File Control Table entry - Name is the file DD name
FIO-*nn*	nth File I/O Area
FWA-*nn*	nth File Work Area

Keyword	Description
ICE-*nn*	nth Interval Control Area
JCA-*nn*	nth Journal Control Area
LLA-*nn*	nth Load List Area
OPF	Optional Feature List
PAM	Page Allocation Map
PCT-*name*	Program Control Table entry - Name is the transaction-ID
PLB	Address of the CICS Partition Lower Boundary Address
PPT-*name*	Processing Program Table entry - Name is the module name
PUB	Address of the CICS Partition Upper Boundary Address
RSA-*nn*	nth Register Storage Area
SIT	System Initialization Table
SYS	Task Control Area (System)
TCA	Task Control Area (User)
TCE	Terminal Control Table Terminal Entry
TCT-*name*	Terminal Control Table entry - Name is the terminal-ID
TIA	Current Terminal Input/Output Area
TIO-*nn*	nth Terminal Input/Output Area
TSA	Temporary Storage Allocation Table

Keyword	Description
TSI-*nn*	nth Temporary Storage Area
TSM	Temporary Storage Map
TUA	Terminal User Area (same as CIA)
TWA	Task Work Area
U24-*nn*	nth 24-bit address User Transaction Storage Area
U31-*nn*	nth 31-bit address User Transaction Storage Area
XCLS	SmartTest Global Exclude CSECT Table
*xx*P	Entry point of the *xx*P CICS NUCLEUS Program, where *xx* are the initials of the Program name (e.g., DCP, FCP)

SmartTest Storage Area Keywords

Keyword	Description
address	Absolute Address. Only valid hexadecimal characters of A through F and 0 through 9 can be entered (e.g., 2C41A0)
MOD-*name*	Load Module for any loaded module in the system (defaults to the qualified load module when entered without the dash)
PGMA	Active program for a test session
PGMQ	Qualified module.program
PGM-*name*	Program for any CSECT of the qualified load module (defaults to the qualified program when entered without the dash)
R0-R15	General Purpose Registers (e.g., R0, R1, ... R15)
AR0-AR15	Access Registers (e.g., AR0, AR1 ... AR15)

COBOL II Keywords

Keyword	Description
BL-*nn*	nth OS/VS COBOL base locator for WORKING-STORAGE
BLF-*nn*	nth COBOL II base locator for files
BLL-*nn*	nth COBOL base locator for the LINKAGE SECTION
BLW-*nn*	nth COBOL II base locator for WORKING-STORAGE
FCB-*nn*	nth COBOL II base locator for file FCB storage

Keyword	Description
FD-*nn*	nth OS/VS COBOL base locator for non-VSAM files
INX-*nn*	nth COBOL index cell
TGT	COBOL Task Global Table
WKS	COBOL WORKING-STORAGE

System/Assembler Keywords

Keyword	Description
ASCB	Address Space Control Block
CDE-name	Contents Directory Entry (same as MOD-name)
CVT	Communication Vector Table
DEB-*nn*	nth Data Extent Block
LLE-*nn*	nth Load List Element
PSW	Program Status Word (formatted) and registers for the last program interrupt
RB-*nn*	nth Request Block
SDWA	System Diagnostic Work Area at the time of an abend
TCB	Task Control Block
TIOT	Task Input/Output Table

9 Action Bar Equivalents To Commands

Command	Pull-down	Action/Option
ADD	Test	Add
ALLOCDEF	Options	Product Allocations
ANALYZE	File	Compile/Analyze
BRANCH	Search	Branch
BREAK	Test	Break
CANCEL	Test	Cancel
COPY	Options	Scratchpad; Copy
DELETE	Options	Scratchpad; Delete
DISPLAY	View	Display
DROP	View	Drop
DUMP	Test	CICS Dump
ENVIRONMENT	File	Setup test environment; Select execution environment
EQUATE	Options	Scratchpad; Equate

Command	Pull-down	Action/Option
EXCLUDE	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Exclude
	View	Exclude
EXECUTE	File	Execute
FIND	Search	String
FINDXTND	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Find
FLOW	Logic	All actions
GO	Test	Go
HELP	Help	All actions
HIGH	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Highlight
KEEP	View	Keep
KEYS	Options	PF Keys
LIST	List	All actions
LOCATE	Search	Line
LPRINT	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Print

Command	Pull-down	Action/Option
LPUNCH	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Punch
MARK	Options	Scratchpad; Mark
MERGE	Options	Scratchpad; Merge
MOVE	Test	Move
NEWCOPY	Test	CICS Newcopy
PARMDEF	Options	Product Parameters
PREF	View	Paragraph X-Ref
PRINTLOG	Options	Log/List/Punch
PRINTLST	Options	Log/List/Punch
PRODLVL	Help	About
QUALIFY	View	Qualify
RENAME	Options	Scratchpad; Rename
RESET	View	Reset
RUN	Test	Run
SAVE	File	Save
SCROLL	Search	Data, Label, Paragraph, String, Subset, Program, Line, Any; Scroll

Command	Pull-down	Action/Option
SET	Options	Modes
SETUP	File	Setup Test Environment
SHOW	View	CICS Show
STEP	Test	Step
STOP	Test	Stop
SUBTRACT	Test	Subtract
TOGGLE	View	Toggle
TRACE	Logic	All actions
UPDATE	File	Edit Pseudo
USING	View	Using
UTILITY	File	AKR Utility
WHERE	Test	Where
ZOOMDATA	View	ZoomData